

Grammaire du Français

Nous avons ainsi travaillé sur la construction d'une grammaire DRTN lexicalisée du français en exploitant les informations de sous-catégorisation et transformationnelles des prédicats du français décrits dans les tables du lexique-grammaire. Notre grammaire est lexicalisée dans le sens où, pour chacun de ces éléments (verbe plein, nom ou adjectif prédicatif ou expression figée), nous décrivons explicitement dans notre grammaire l'ensemble des réalisations possibles de chaque constituant syntaxique dont il peut être le noyau.

Ces constituants peuvent être des phrases complètes, des propositions infinitives, participiales, des propositions relatives dans lesquelles un élément a été extrait, des groupes nominaux complexes dont la tête est un nom prédicatif, etc.

Comme nous le verrons, nous avons pu réduire la complexité d'une telle description systématique par l'écriture manuelle d'une meta-grammaire à partir de laquelle nous générons l'ensemble de ces descriptions lexicalisées en fonction du contenu des tables du lexique-grammaire.

4.5.1 Construction semi-automatique

Notre grammaire DRTN lexicalisée finale, qui est utilisée pour l'analyse syntaxique, est composée de l'ensemble des grammaires spécialisées pour chacun des éléments prédicatifs que nous traitons. Cette grammaire est générée automatiquement à partir d'un ensemble de meta-grammaires que nous avons

construites manuellement. Une meta-grammaire est composée d'un ensemble de graphes paramétrés associés à une table du lexique-grammaire. L'utilisation des graphes paramétrés fonctionne sur le même principe que le système de génération de grammaires par graphes patrons inclus dans les logiciels INTEX et Unitex (même si la nomenclature est un peu différente). Ce système de graphe patron a été repris dans plusieurs travaux exploitant les tables du lexique-grammaire pour l'analyse de textes ([Senellart, 1999] [Paumier, 2003a] par exemple) ou des recherches linguistiques sur corpus [Camugli Gallardo et Blanc, 2006].

Dans le cas de notre grammaire, chaque graphe paramétré décrit un constituant syntaxique de la grammaire dont le prédicat est une variable qui sera instanciée durant la phase de lexicalisation de la grammaire. Nous pouvons voir la meta-grammaire d'une table (c'est-à-dire l'ensemble des graphes associés à cette table) comme la grammaire de toutes les constructions syntaxiques de cette table, même si chaque construction n'est pas compatible avec toutes les entrées. Chaque chemin de cette grammaire est identifié par un paramètre qui réfère à la propriété qui lui correspond dans la table. Un paramètre a le format suivant : @X@, où X est l'intitulé d'une colonne de la table. Durant la phase de lexicalisation, le processus de génération de la grammaire construit pour chaque entrée de la table, une grammaire spécialisée dans laquelle seuls les chemins correspondant aux propriétés syntaxiques acceptées par cette entrée sont conservés. Lorsqu'une propriété identifiée par un paramètre est acceptée par l'entrée (c'est-à-dire lorsque l'intersection de la ligne et de la colonne considérées contient un +), le paramètre est remplacé par la chaîne vide dans le grammaire générée. La chaîne vide assure la continuation du chemin dans l'automate et donc que la construction est conservée. Quand la propriété n'est pas acceptée, la transition est supprimée de la grammaire ce qui a pour effet de bloquer l'ensemble des chemins passant par cette transition. Certaines propriétés ne sont pas de nature booléenne, mais contiennent des valeurs textuelles (par exemple la valeur d'une préposition), dans ce cas, le paramètre est remplacé dans la grammaire lexicalisée par la valeur de cette propriété. Il est également possible de nier un paramètre : le paramètre @!X@ signifie que le chemin identifié par ce paramètre sera conservé uniquement si la propriété d'intitulé X n'est pas acceptée par l'entrée considérée. Enfin, le paramètre spécial @%id@ sera remplacé au moment de la lexicalisation par le numéro de ligne de l'entrée considérée. Ce paramètre est utile pour identifier de façon unique une entrée d'une table du

lexique grammairale, même lorsque cette table contient plusieurs entrées pour un même lemme.

Par exemple, la figure 4.35 présente un graphe paramétré associé à la table 6. Cette table comporte la description des verbes à structure transitive simple avec complément direct phrastique, c'est-à-dire les verbes (comme *admettre*) qui entrent dans la construction $N0 V N1$, avec $N1 = \text{que } P + N$:

Luc a finalement admis (son erreur + que Lea l'a quitté).

Nous décrivons dans notre graphe les formes de base pour les verbes de cette table². Les différents chemins de la grammaire sont identifiés par des paramètres qui réfèrent aux propriétés de la table leur correspondant. Par exemple, le paramètre @N0=Nhum@ réfère à la colonne dans la table indiquant si le verbe admet un sujet humain, le paramètre @NOV@ réfère à la colonne qui indique si le verbe admet l'ellipse de son objet direct. De même, le paramètre @entry@ à valeur textuelle fait référence à la colonne dans la table où est donné le lemme de l'entrée. Le graphe de la figure 4.36 montre le résultat de la lexicalisation de la grammaire pour le verbe *admettre*.

Contrairement à ce que laisse penser notre exemple, nous ne construisons pas, durant la lexicalisation de la meta-grammaire d'une table, un graphe lexicalisé différent pour chaque combinaison de graphe paramétré et d'entrée de la table. Dans la pratique, nous aplatissons au préalable chaque graphe paramétré principal décrivant un constituant syntaxique de la grammaire, de manière à ce qu'il soit décrit par un unique automate sans appel à des sous-graphes (cf. section 3.4). Puis, à partir de chacun de ces automates paramétrés, nous construisons sa version lexicalisée constituée de l'union des descriptions de ce constituant syntaxique lexicalisées pour chacune des entrées de la table. La grammaire générale est ensuite obtenue en construisant chaque constituant syntaxique par l'union des constituants lexicalisés de chaque table. Nous optimisons également notre grammaire, en effectuant

²Dans la pratique, nous avons fait la description du verbe principal et de ses arguments $N0$ et $N1$ dans des sous-graphes distincts de manière à pouvoir les réutiliser directement pour la description d'autres types de constructions. Nous avons inclus ces descriptions dans le graphe principal pour notre exemple de manière à illustrer le procédé de lexicalisation de la meta-grammaire.

pour chacun de ces constituants, les opérations d'émondation, de suppression des transitions étiquetées par le mot vide, de déterminisation et de minimisation telles que nous les appliquons pour l'optimisation des grammaires WRTN (cf. section 3.4).

Notons que nous aurions pu mettre en place un système permettant de générer, à partir d'une unique meta-grammaire, l'ensemble des constructions lexicalisées de chaque entrée prédicative décrite dans les tables du lexique-grammaire. C'est d'ailleurs l'approche qui a été choisie dans [Paumier, 2003a], par l'utilisation d'une super table. Cependant, un tel procédé n'est pas trivial à mettre en place puisqu'il existe de nombreuses propriétés qui sont spécifiques à certaines tables uniquement. Ainsi, le fait de représenter toutes ces constructions dans une unique meta-grammaire alourdirait considérablement cette dernière au risque de la rendre difficile à lire et à maintenir. De plus, certaines propriétés peuvent avoir un même intitulé mais une signification différente en fonction de la table dans laquelle elle sont encodées³. Pour toutes ces raisons, nous avons décidé de construire manuellement une meta-grammaire différente pour chaque table. Après avoir expérimenté avec ce système, nous nous sommes aperçu que nous obtenons de nombreuses redondances dans les descriptions des différentes meta-grammaires, ce qui cause également des problèmes pour la maintenance de la grammaire dans son ensemble. Nous en sommes donc arrivé à la conclusion qu'il serait sans doute souhaitable de faire un compromis entre ces deux approches, en mettant en place un système, que nous n'avons pas eu le temps de développer, dans lequel une meta-grammaire serait associée à un ensemble de tables qui auraient certaines propriétés en commun, comme par exemple le nombre des actants acceptés par les prédicats.

³Ce problème n'en est plus un aujourd'hui, puisqu'une révision complète des intitulés des colonnes a été effectuée récemment.

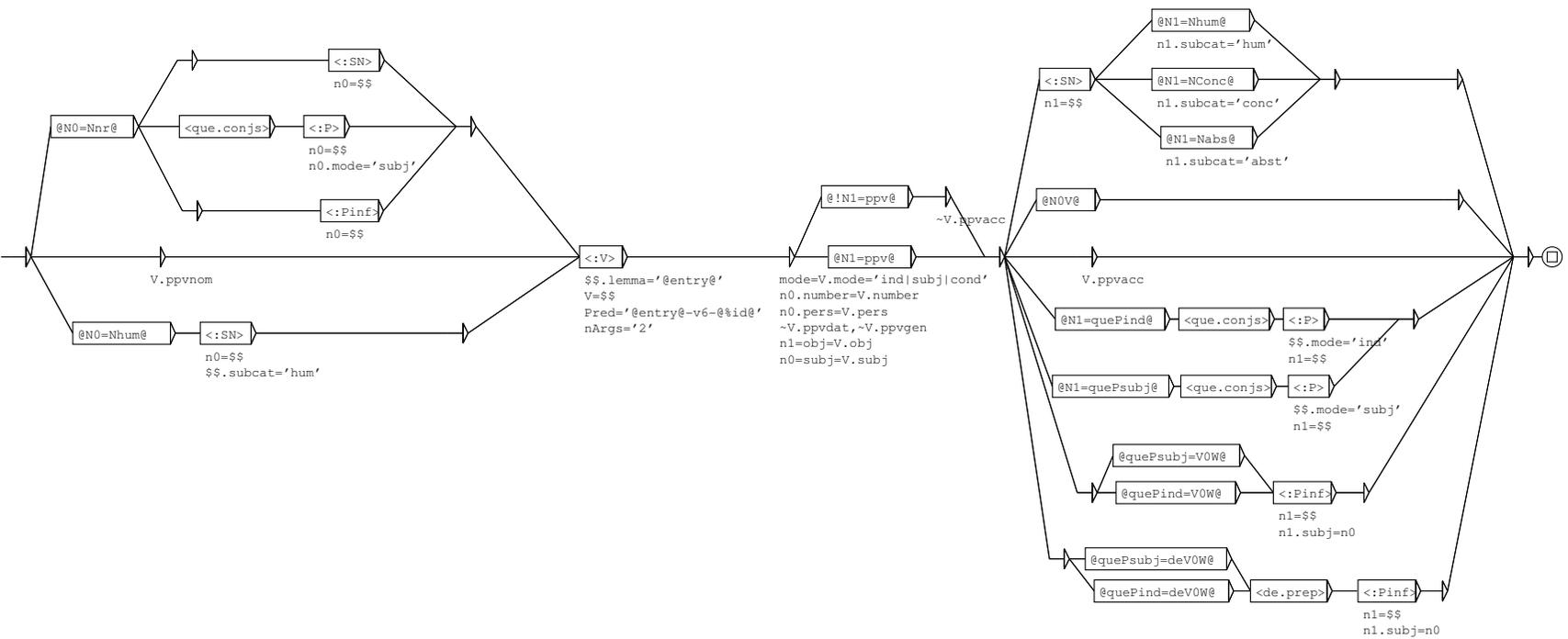


FIG. 4.35 – Exemple de graphe paramétré

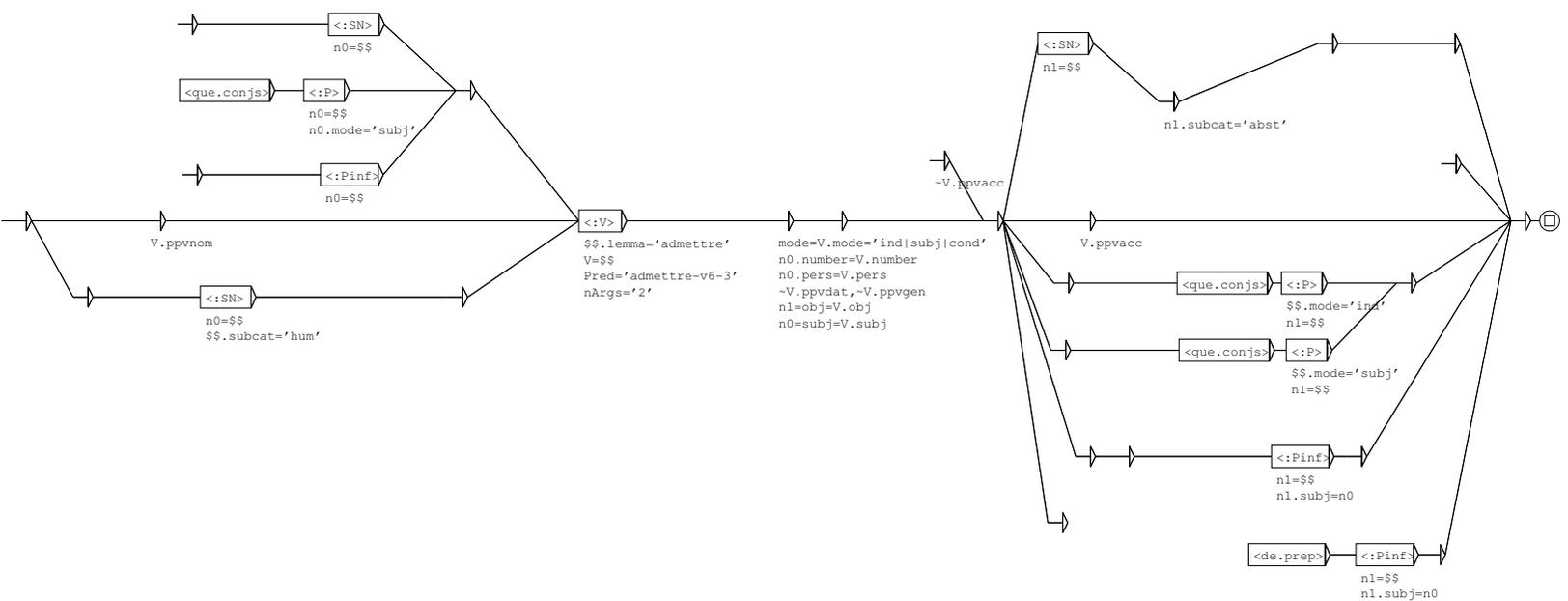


FIG. 4.36 – Spécialisation pour le verbe admettre

4.5.2 Remarques sur l'analyse

Nous avons implémenté un nouvel analyseur spécialisé pour les grammaires d'unification DRTN. Théoriquement, nous aurions pu implémenter notre analyseur DRTN comme une routine de post-traitement branchée à notre analyseur pour grammaire WRTN présenté dans le chapitre précédent. Cette routine aurait été appelée après l'analyse grammaticale par la grammaire WRTN support, et aurait vérifié les contraintes d'unification et construit les structures de traits résultat à partir de la forêt d'arbres décorés obtenue comme résultat de l'analyse. C'est d'ailleurs cette approche qui a été choisie pour l'implémentation de l'analyseur LFG efficace SXLFG [Boullier et Sagot, 2005] qui fonctionne en deux passes : une première passe effectue la reconnaissance par la grammaire algébrique support et une seconde passe vérifie les contraintes d'unification sur les arbres de dérivation obtenus. Cependant, cette approche ne nous a pas semblé viable dans notre cas, du fait de la forme de notre grammaire qui est fortement lexicalisée au niveau des contraintes d'unification. En raison des ambiguïtés, la grammaire contient un grand nombre de chemins qui reconnaissent les mêmes séquences en entrées et qui diffèrent seulement par les contraintes d'unification spécialisées pour chaque entrée lexicale. Ainsi, nous avons adapté notre algorithme d'analyse pour les grammaires DRTN de manière à vérifier les contraintes d'unification présentes dans les sorties et à construire les structures de traits associées à chaque constituant syntaxique durant l'analyse. Ceci nous permet de stopper plus tôt la reconnaissance d'analyses qui échouent du fait que les structures de traits qui leur sont associées ne peuvent pas s'unifier. Nous obtenons, comme résultat de notre analyseur, une forêt d'arbres de dérivation ; à chacun de ces arbres est associée une structure de traits bien formée. L'arbre syntaxique rend compte du découpage syntagmatique de la phrase et nous représentons dans la structure de traits associée différentes relations grammaticales qui relient ces constituants.

Dans le cas de notre grammaire, nous nous sommes essentiellement intéressé à identifier dans les textes les prédicats sémantiques et leurs arguments. Nous avons ainsi adopté une convention de notation particulière pour représenter ce type de relations dans la structure de traits. Les prédicats sont représentés par un trait *Pred* ; la valeur de ce trait identifie de façon unique une entrée d'une table du lexique-grammaire : celle-ci consiste en la concaténation du

lemme de l'entrée, suivi du nom de la table à laquelle elle appartient suivi par son identifiant dans cette table (i.e. le numéro de la ligne dans laquelle l'entrée est décrite). Par exemple, la valeur du trait **Pred** pour l'emploi du verbe *admettre* décrit dans la table 6 est **admettre-v6-3** (cf. l'exemple de la figure 4.36). Nous accompagnons l'attribut **Pred** par un attribut **nArgs** qui doit être présent au même niveau dans la structure de traits. La valeur de cet attribut indique le nombre d'arguments attendus par ce prédicat (voir l'équation **nArgs='2'** dans notre exemple de la figure 4.36). Les valeurs de ces arguments sont stockées, lorsqu'ils sont présents, dans les traits **n0**, **n1**, etc.

Nous représentons également certaines relations de modifieur à modifié dans notre grammaire. Par exemple, nous analysons une phrase subordonnée comme un modifieur de la phrase principale, ou encore une phrase relative ou participiale comme modifieur du syntagme nominal auquel elle est accolée. Tous ces modifieurs sont stockés dans l'attribut **modifs** de la structure de traits décrivant le constituant qu'ils modifient. Cet attribut prend pour valeur un ensemble de structures de traits.

Par exemple, étant donnée la phrase suivante :

L'homme qui parlait à Lea est parti.

Nous obtenons comme résultat de son analyse avec notre grammaire, la structure de traits suivante, que nous avons simplifiée en conservant uniquement les informations qui nous intéressent :

$$\left[\begin{array}{l} \text{Pred : partir-v2-61} \\ \text{nArgs : 3} \\ \text{n0 : } \left[\begin{array}{l} \text{head : (1) [form : homme]} \\ \text{modifs : } < \left[\begin{array}{l} \text{Pred : parler-v15-45} \\ \text{nArgs : 3} \\ \text{n0 : (1)} \\ \text{n2 : [head : [form : Lea]]} \end{array} \right] > \end{array} \right] \end{array} \right]$$

FIG. 4.37 – Structure de traits (simplifiée) résultat de l'analyse

Notons que la valeur de l'attribut `nArgs` ne correspond pas au nombre d'arguments essentiels qui sont effectivement présents dans l'énoncé analysé ; cette valeur correspond au nombre d'arguments qu'accepte le prédicat dans la phrase canonique qui a servi à sa classification dans les tables du lexique-grammaire (dans laquelle tous ses arguments jugés comme essentiels sont réalisés). Par exemple, le prédicat principal de notre phrase analysée a été identifié comme étant le verbe *partir* dans son emploi défini dans la table 2. Ce verbe accepte dans sa forme la plus longue trois arguments, dont un complément locatif et une proposition infinitives qui peuvent être réalisés tous les deux dans le même énoncé :

Luc est parti à la boulangerie chercher du pain.

Dans notre exemple, seul l'argument `n0` (le sujet) est réalisé syntaxiquement.

De manière à faciliter la lecture des résultats d'analyse, nous avons implémenté un extracteur de relations entre prédicat et arguments. Etant donnée une structure de traits dont la forme suit nos conventions de notation, notre programme en extrait, par un simple parcours récursif de la structure, toutes ces relations et les présente sous la forme de prédicats logiques. Ainsi, nous obtenons comme résultat de l'analyse de notre phrase les deux formules suivantes :

```
partir-v2-61(homme, _, _)
parler-v15-45(homme, _, Lea)
```

4.5.3 Constituants syntaxiques généraux

Outre les descriptions lexicalisées, qui sont générées à partir de notre meta-grammaire, nous avons également fait la description de constituants généraux non lexicalisés qui sont utilisés dans la plupart des constituants de notre grammaire. Il s'agit, entre autres, des constituants `Ins`, pour la description des adverbes d'une phrase, `SN` pour la description des syntagmes nominaux, `SA` pour celles des syntagmes adjectivaux, et `V` pour la description du noyau verbal. Notons que ces descriptions générales peuvent également invoquer des

descriptions lexicalisées. Par exemple, notre grammaire des SN reconnaît des formes générales de groupes nominaux, mais intègre également les descriptions lexicalisées des groupes nominaux à tête prédicative qui sont générées à partir de la meta-grammaire des tables de noms. Cette même grammaire décrit la possibilité de modifier le nom tête par une proposition relative qui est elle-même décrite par la grammaire lexicalisée de son prédicat principal. Il en est de même pour la grammaire des adverbes, qui comprend par exemple la description des phrases introduites par une conjonction de subordination. Ainsi, toutes ces descriptions générales forment une sorte de liant entre les descriptions lexicalisées décrites dans la meta-grammaire, et permettent de ce fait d'obtenir une grammaire cohérente capable d'analyser des énoncés complexes comprenant plusieurs propositions enchassées.

Nous présentons ici plus en détail notre grammaire SN décrivant les syntagmes nominaux et notre constituant V qui décrit le noyau verbal d'une phrase (conjuguée, infinitive ou participiale).

Constituant SN

Le graphe de la figure 4.38 présente notre grammaire `SNsimple`, pour la description des syntagmes nominaux simples, c'est-à-dire non coordonnées. À gauche, nous décrivons la tête du syntagme nominal, qui peut être réalisée soit par un pronom tonique (*moi, lui-même*, etc.), indéfini (*personne, quelqu'un*, etc.) ou démonstratif (*celui-ci, ceux*, etc.), soit par un nom propre décrit dans le constituant `Npr`, soit par un autre groupe nominal non récursif décrit dans le constituant `chunk`. Dans tous les cas, nous unifions la structure de traits associée à ce constituant avec l'attribut `head` du constituant englobant (avec l'équation `head=$$`), et nous faisons remonter les informations indiquant son genre, son nombre, sa personne, sa sous-catégorie sémantique et son lemme au niveau du SN (équations `^gender`, `^number`, `^pers`, `^subcat` et `^lemma`).

Le constituant `chunk` est présenté dans la figure 4.39 ; ce graphe décrit le syntagme nominal jusqu'à son nom tête (c'est-à-dire précédé d'un ou plusieurs déterminants et éventuellement d'un groupe adjectival), les éventuels modificateurs à droite de celui-ci sont décrits dans le graphe principal `SNsimple`. Nous ne présentons pas les graphes `Det` et `Adjg`, décrivant respectivement les dé-

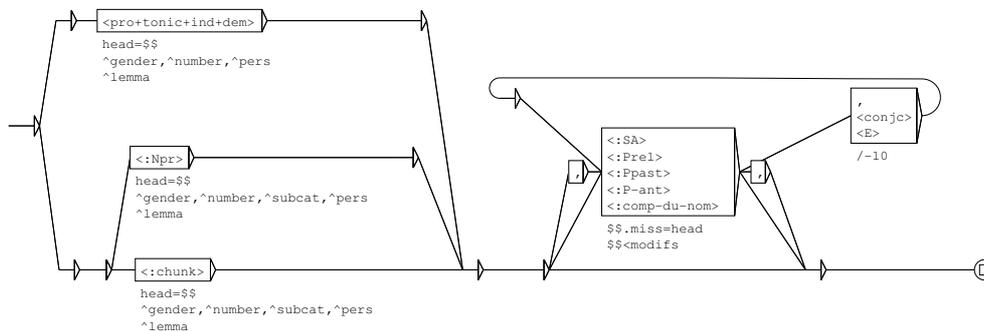


FIG. 4.38 – Constituant SNsimple

terminants et groupes adjectivaux qui peuvent apparaître à gauche du nom. Ces graphes sont pour l’instant très simples, nous avons cependant décrit ces éléments dans des sous-graphes distincts de manière à pouvoir facilement les remplacer par des descriptions plus complètes. La pondération de poids 10 dans notre grammaire sur la transition identifiant le nom tête comme un nom composé permet de préférer cette analyse aux analyses identifiant le mot de tête comme un mot simple (de poids 0). En effet, beaucoup de noms composés français (comme *pomme de terre* ou *carte bleue* par exemple) ont des structures internes (de type NA ou NprepN) qui les rendent difficiles à distinguer des mots simples suivis par un modifieur ; cette heuristique nous permet donc de réduire le nombre d’ambiguïtés d’analyse en préférant les analyses figées aux analyses compositionnelles.

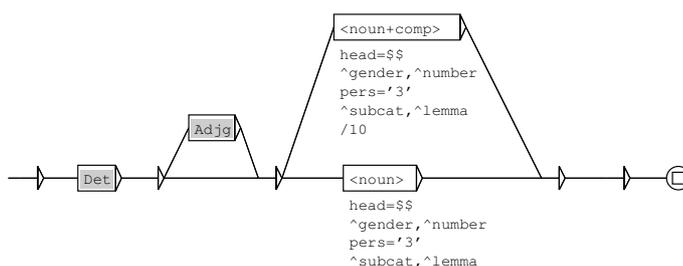


FIG. 4.39 – Constituant chunk

Dans la partie droite du graphe `SNsimple`, nous décrivons les modifieurs qui peuvent apparaître à droite du nom tête : il peut s'agir d'un syntagme adjectival (constituant `SA`), d'une phrase relative (constituant `Pre1`), d'une participiale passé ou présent (constituants `Ppast` et `P-ant`) ou d'un complément du nom (constituant `comp-du-nom`). Nous ajoutons la structure de traits associée à ce constituant à la liste des modifieurs du syntagme nominal avec l'équation `$$<modifs`. Avec l'équation `$.miss=head`, nous unifions le trait `miss` du modifieur avec la structure de traits associée à la tête de notre syntagme nominal. Nous utilisons ce type d'équation dans différentes parties de notre grammaire afin de décrire les différents phénomènes d'extraction en transmettant l'information sur l'élément manquant dans le constituant privé de cet élément par l'intermédiaire du trait `miss`.

Par exemple, la figure 4.40 présente notre graphe `Pre1` décrivant les propositions relatives. Nous y décrivons, entre autres, la possibilité d'introduire la proposition relative par le pronom *qui* ou *lequel* (ou une de ses formes fléchies) ; dans ce cas, le pronom est suivi par une phrase privée de son sujet décrit par le constituant `P-Nnom`. L'équation `^miss` sous la boîte étiquetée par l'appel à `P-Nnom` permet de propager le trait `miss` dans la description de ce constituant. Notre grammaire `P-Nnom` est présentée dans la figure 4.41, ce graphe consiste simplement en des appels à des sous-graphes sur l'ensemble des grammaires lexicalisées pour le constituant `P-Nnom` que nous avons décrit dans notre meta-grammaire. Nous donnons des exemples de ces descriptions dans la section 4.5.4 qui présente des extraits de notre meta-grammaire.

Ce même trait pourra éventuellement être propagé dans d'autres propositions enchassées nous permettant ainsi de formaliser les dépendances à distance à la manière du trait *Slash* utilisé en HPSG. Par exemple, notre grammaire reconnaît, avec ce mécanisme de propagation, la séquence suivante comme un syntagme nominal :

la fille que Luc dit avoir reconnue.

et lui associe l'analyse suivante, dans laquelle le nom *fille* antéposé, dont la structure de traits a été propagée dans les propositions enchassées par le biais du trait `miss`, a été identifié comme l'argument *n1* du verbe *reconnaître* :

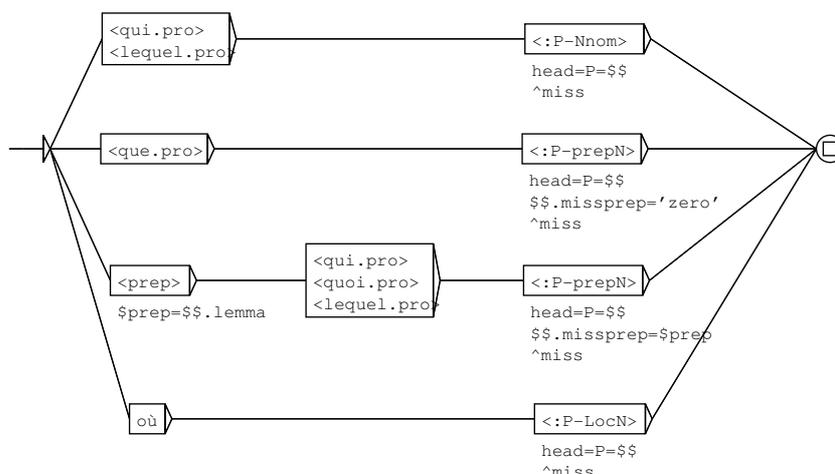


FIG. 4.40 – Constituant Prel

dire-v9-125(Luc, reconnaître-v32r2-352(Luc, fille), _)

Nous présentons, pour exemple, dans la figure 4.42 le graphe *comp-du-nom* qui décrit les compléments du nom. Ce graphe, très général, décrit ce type de modifieur comme un syntagme nominal introduit par une préposition. A l'aide des équations sur les traits, nous associons à ce type de construction une analyse dans laquelle le prédicat est la préposition introductrice qui accepte deux arguments qui sont respectivement le nom tête du syntagme nominal (unifié par l'intermédiaire de l'attribut *miss*) et le nom tête du complément du nom. Ainsi, par exemple, les syntagmes nominaux :

la pomme sur la table
le fils du facteur
les jardins de Paris
la boîte de trombones
la barre de métal

recevront les analyses :

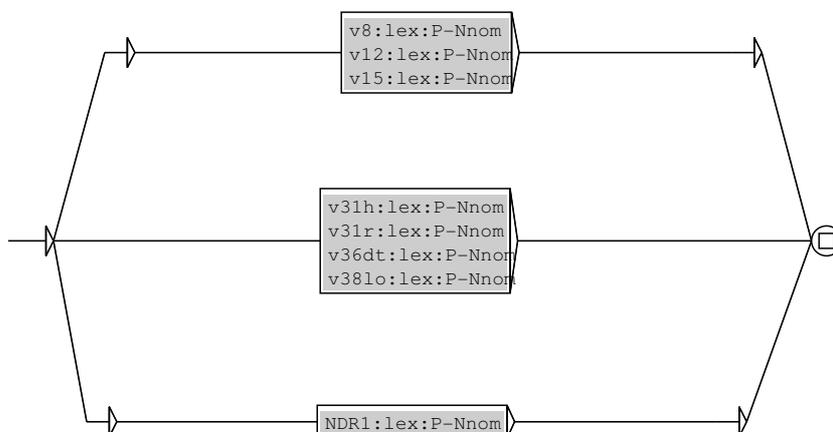


FIG. 4.41 – Constituant P-Nnom

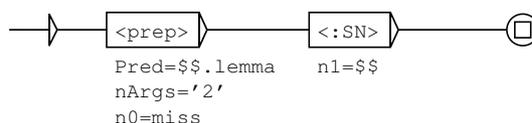


FIG. 4.42 – Constituant comp-du-nom

sur(pomme, table)
 de(fil, facteur)
 de(jardins, Paris)
 de(boîte, trombones)
 de(barre, métal)

Comme nous le voyons, une même préposition peut introduire différents types de relations entre le modifieur et son modifié ; nous n'avons pas cherché pour l'instant à affiner cette description, puisque nous n'avons pas les moyens de résoudre ce problème dans le cas général⁴. Cependant, dans le cas où le nom

⁴Afin de décider de la nature de la relation qui est introduite par la préposition, il nous semble nécessaire d'avoir à notre disposition une ontologie fine dans laquelle seraient classés les deux substantifs qui interagissent dans cette relation.

tête du syntagme nominal est un nom prédicatif décrit dans notre grammaire, nous pouvons obtenir des analyses plus fines dans lesquelles les compléments du nom sont reconnus comme des arguments du prédicat en fonction de leur préposition introductrice (cf. section 4.5.4).

Enfin, pour terminer la description de notre grammaire des SN, notons que nous avons décrit la possibilité de reconnaître un nombre non borné de modifieurs à droite du nom tête en bouclant sur ces descriptions dans notre grammaire (figure 4.38). Nous avons cependant pondéré cette boucle par un poids négatif de -10, de manière à réduire les ambiguïtés d'attachement prépositionnel. Cette pondération négative nous permet d'associer des poids moins élevés aux analyses dans lesquelles la distance entre le modifieur et son modifié est plus importante. Cette heuristique nous permet ainsi, dans le cas où le nom de tête du SN est suivi par une séquence de modifieurs, de préférer les analyses où chaque modifieur est attaché au nom le plus proche à sa gauche. Ainsi, si nous prenons la séquence suivante qui est naturellement ambiguë :

Le fils du facteur que Luc nous a dit avoir reconnu

Cette séquence reçoit à l'issue de l'analyse par notre grammaire une unique analyse sur les deux possibles, dans laquelle Luc reconnaît le facteur et non le fils de ce dernier :

de(fil, facteur)
dire-v9-125(Luc, reconnaître-v32r2-352(Luc, facteur), nous)

Nous avons recouru à cette heuristique pour des raisons purement pratiques ; en effet, sans celle-ci, l'ambiguïté d'attachement prépositionnel nous créait un nombre tellement important de résultats d'analyse (de l'ordre de mille analyses par phrase) que ces résultats étaient inexploitablement en pratique. Nous avons ainsi préféré, quitte à perdre l'analyse correcte, limiter de manière arbitraire le nombre de résultats d'analyse pour les cas d'ambiguïté que nous ne savons pas résoudre a priori. Nous avons néanmoins conservé la boucle dans notre grammaire, pour analyser des cas où un modifieur ne peut pas être attaché au nom le plus proche à sa gauche, comme par exemple dans la séquence :

d'après les conclusions de ce rapport, approuvées par le conseil européen

Ici, la phrase participiale de verbe *approuver* ne peut pas être attachée au nom *rapport* du fait que ces deux mots ne vérifient pas les contraintes d'accord en genre et en nombre spécifiées dans notre grammaire. Ainsi, nous obtenons à l'issue de l'analyse, le résultat suivant dans lequel l'argument n1 du verbe *approuver* est bien le mot *conclusions* et non *rapport* :

```
de(conclusions, rapport)
approuver-v12-10(conseil européen, conclusions)
```

Non terminal V

Nous présentons dans la figure 4.43 le graphe principal du constituant syntaxique V, qui décrit le noyau verbal d'une phrase (conjuguée, infinitive ou participiale), c'est-à-dire son verbe principal éventuellement modifié par des auxiliaires de temps, modaux ou aspectuels. Nous décrivons également dans ce constituant la possibilité d'apparition de pronoms clitiques à gauche du verbe (par un appel au sous-graphe C1 présenté dans la figure 4.44).

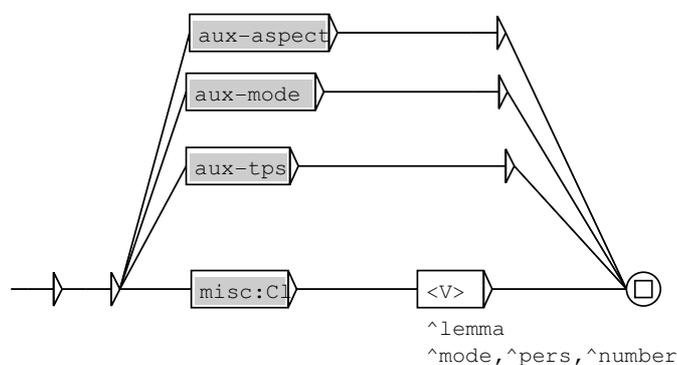


FIG. 4.43 – Constituant V

L'ordre d'apparition de ces pronoms est assez contraint en français, comme le montre notre description :

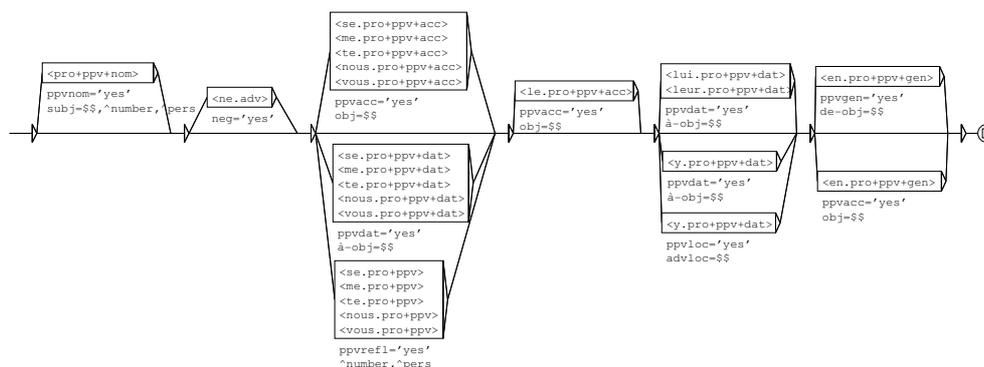


FIG. 4.44 – Sous-graphe C1

- d’abord le pronom sujet (*je, tu, il, etc.*) ; dans ce cas, nous positionnons à *yes* la valeur du trait `ppvnom` du constituant `V` et nous unifions la valeur du trait `subj` avec la structure de trait associée au pronom sujet ; les équations `^pers` et `^number` permettent de vérifier l’accord en genre et en nombre entre le verbe et son sujet.
- il est ensuite possible de voir apparaître un des pronoms personnels *me, te, se, nous* ou *vous*. Le pronom peut recevoir alors 3 analyses distinctes :
 - soit il s’agit d’un pronom accusatif : *Luc nous observe*. Dans ce cas, nous positionnons la valeur du trait `ppvacc` à *yes* et nous unifions la structure de trait associée à ce pronom avec le trait `trait obj` du constituant `V` (équation `obj= $$`).
 - soit il s’agit d’un pronom datif : *Luc nous a offert un verre*. Dans ce cas, nous positionnons la valeur du trait `ppvdat` à *yes* et nous unifions la structure de trait associée au pronom avec le trait `à-obj` du constituant `V`.
 - soit il s’agit d’un pronom réfléchi : *Luc s’endort*. Dans ce cas, il faudrait distinguer quatre cas : le *se* figé (*Luc s’enquiert de la santé de Lea*) ; le *se* grammatical (*la branche se casse*) ; et le *se* résultat de la pronominalisation d’un complément direct (*Luc se prend pour Napoleon*) ou indirect (*Luc s’envoie un message sur son répondeur*). Nous ne faisons pas pour l’instant cette distinction dans notre grammaire et nous positionnons la valeur du trait `ppvrefl` à *yes* dans tous ces cas, en considérant que

le pronom réfléchi fait parti du prédicat verbal. Nous distinguons de cette manière, le verbe intransitif *s'endormir* (décrit dans la table 35R) du verbe transifif *endormir* (table 4), prédicat de la phrase : (*Luc+le discours de Luc*) nous endort.

- les pronoms préverbaux *le, la, les* et les pronoms *lui, leur* ou *leurs* reçoivent la même analyse que les pronoms personnels respectivement accusatifs et datifs.
- le pronom préverbal *y* peut recevoir deux analyses : soit il pronominalise un complément datif : *Luc y pense tous les jours* ; soit il pronominalise un adverbe le plus souvent locatif : *le resto U, Luc y mange tous les jours*. Cet adverbe peut également être un argument essentiel d'un prédicat à complément locatif : *Luc y met son dentifrice*. Dans ce dernier cas, nous positionnons la valeur de l'attribut *ppvloc* à *yes* et nous unifions le trait *advloc* avec la structure de traits associée à ce pronom.
- Enfin, la forme *en* peut également recevoir deux analyses : il peut pronominaliser un complément introduit par la préposition *de* (*Luc en parle*), mais également un complément accusatif comme dans la phrase *Luc en a déjà mangé*.

Dans sa forme la plus simple, un noyau verbal consiste en un verbe éventuellement précédé par des pronoms clitiques (cf. figure 4.43). Dans ce cas, nous faisons remonter au niveau du constituant *V*, les informations sur le lemme du verbe, son mode, sa personne et son nombre ainsi que les analyses sur les pronoms clitiques comme vu précédemment.

Nous décrivons dans des sous-graphes, la possibilité de modifier le verbe par un auxiliaire de temps (sous-graphe *aux-tps*, figure 4.45), un auxiliaire de mode (graphe *aux-mode*, figure 4.46) ou un auxiliaire aspectuel (graphe *aux-aspect*, figure 4.47).

Tous ces graphes ont la même forme générale : ils décrivent l'auxiliaire suivi par un constituant *V* soit au participe passé pour les auxiliaires de temps *être* et *avoir* soit à l'infinitif pour les autres auxiliaires. Le mode du constituant *V* courant est hérité du mode de conjugaison de l'auxiliaire, il en est de même pour les informations de nombre et personne (équations $\hat{\text{mode}}, \hat{\text{number}}, \hat{\text{person}}$). Le lemme du verbe principal, quant à lui, est hérité du constituant *V*, fils du constituant courant ; il en est de même pour les analyses issues de la présence de pronoms clitiques (excepté pour le pronom

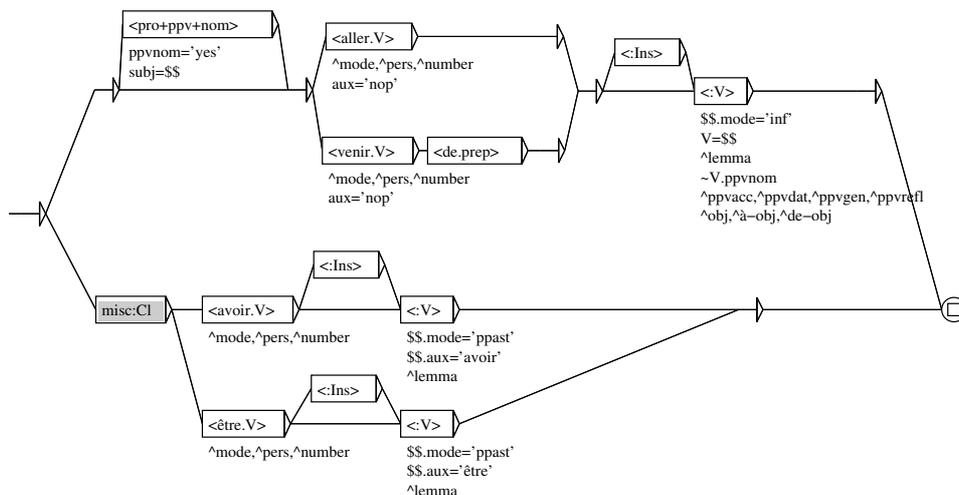


FIG. 4.45 – Sous-graphe aux-tps

sujet qui ne peut être réalisé qu'à gauche du verbe auxiliaire). Enfin, la reconnaissance récursive d'un constituant V à droite de l'auxiliaire, nous permet d'analyser des constructions où le verbe principal est modifié par plusieurs auxiliaires comme la séquence :

(Luc) semble avoir été en train de travailler

qui est analysée par notre grammaire comme un constituant V de lemme *travailler* et conjugué à l'indicatif.

La liste des verbes auxiliaires que nous avons décrits dans notre grammaire ne prétend pas être exhaustive ; en fait, nous nous sommes contenté pour l'instant de reprendre les exemples de ce type de verbes utilisés pour la présentation de la grammaire LTAG pour le français [Abeillé, 2002]. Il s'agit essentiellement de verbes décrits dans la table 1 du lexique-grammaire, c'est-à-dire les verbes à un complément de la forme d'une proposition infinitive qui n'a pas de source complétive attestée. Toutes les entrées de cette table ne peuvent pas pour autant être considérées comme des auxiliaires modaux

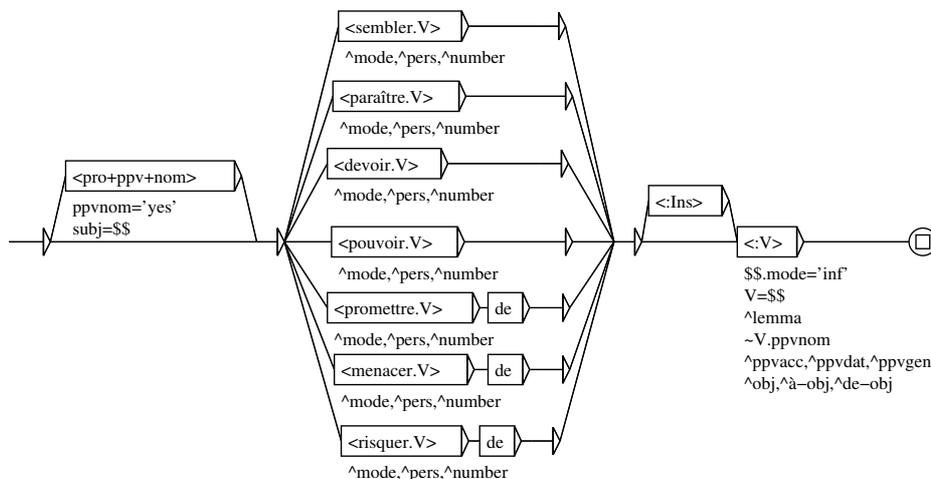


FIG. 4.46 – Sous-graphe aux-mode

ou aspectuels puisque certains de ces verbes (comme par exemple *hésiter à*, *oser*, *oublier de*, etc.) sélectionnent des sujets animés uniquement. Nous considérons ce type de verbes comme des prédicats à deux arguments (le sujet et la proposition infinitive), tels que le premier argument est nécessairement coréférent au sujet du second argument phrastique. En contrepartie, les verbes que nous avons sélectionnés dans notre grammaire n'imposent pas de restriction particulière sur la nature du sujet de la phrase, mais certaines restrictions peuvent néanmoins être imposées par le verbe principal :

*(il + *Luc) pleut*
*(il + *Luc) menace de pleuvoir*

En ce sens, nous les considérons comme des prédicats unaires à argument phrastique. A plus long terme, nous envisageons de refaire la grammaire des auxiliaires en la générant automatiquement à partir d'un sous-ensemble de la table 1 de laquelle nous aurions sélectionné les entrées que nous aurions clairement identifiées comme des auxiliaires de mode ou d'aspect.

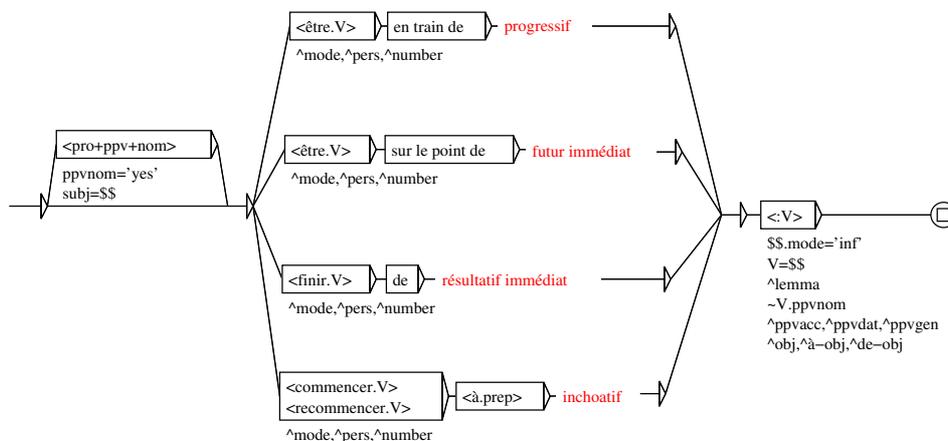


FIG. 4.47 – Sous-graphe aux-aspect

4.5.4 Exemples de tables

La construction de la grammaire complète du français est un long travail. Nous nous sommes pour l'instant concentré sur la syntaxe des verbes simples du français. Pour chacune de ces tables, nous n'avons pas nécessairement fait la description exhaustive de toutes les constructions. D'une façon générale, nous avons procédé de la manière suivante : nous faisons dans un premier temps, pour chaque table, la description des formes canoniques de ses entrées ; puis nous complétons notre grammaire au fur et à mesure que nous rencontrons de nouvelles constructions manquantes lors de son application pour l'analyse de texte. A l'heure actuelle nous avons ainsi entamé la conversion 18 tables de verbes (sur les 60 existantes). Nous avons également intégré la conversion de deux tables de noms prédicatif dans notre grammaire (les tables NDR1 et NAPE).

Nous montrons dans les pages qui suivent des extraits de notre meta-grammaire que nous pensons représentatifs des différents phénomènes de syntaxe que nous sommes parvenu à formaliser et analyser. Plus précisément, nous présentons les meta-grammaire de 3 tables : la table 36DT qui décrit les

verbes à deux compléments non phrastiques ; la table 12 des verbe à un complément phrastique et la table NDR1 de noms prédicatifs à verbes support *donner*.

Table 36DT

La table 36DT regroupe les verbes à trois arguments non phrastiques : un sujet, un complément d’objet direct et un complément datif humain introduit par la préposition *à*. Cette table contient également quelques verbes pour lesquels le second complément est introduit par la préposition *de* :

Paul a hérité cette pendule de son oncle

Afin de simplifier l’écriture de notre meta-grammaire, nous avons scindé la table en deux en fonction de la préposition introductrice. Nous présentons ici la meta-grammaire des verbes dont le second complément est introduit par *à*.

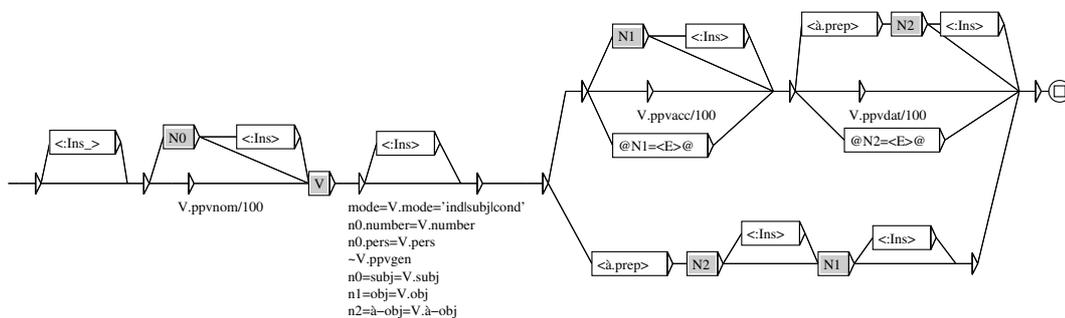


FIG. 4.48 – Formes de base pour la table 36DT

La figure 4.48 décrit les constructions canoniques des verbes de la table (de la forme : *N0 V N1 à N2*). Les équations sur les traits nous permettent entre autres :

- d’imposer que le verbe est conjugué à un temps fini ($mode='ind|subj|cond'$) ;

- de vérifier l'accord en nombre et en personne entre le verbe et son sujet ($n0.number=V.number$ et $n0.pers=V.pers$);
- et d'identifier les arguments du verbe en fonction de leur position syntaxique ($n0=subj$, $n1=obj$ et $n2=\grave{a}-obj$).

Les arguments du verbe peuvent être réalisés syntaxiquement soit sous la forme d'un groupe nominal (décrit dans un sous-graphe de structuration dédié) soit par un pronom clitique, mais pas par les deux simultanément puisque dans ce cas la structure de traits ne pourrait pas s'unifier avec les deux valeurs qui sont incompatibles.

Nous avons isolé la description du verbe et de ses arguments dans des sous-graphes (présentés dans les figures 4.49 et 4.50) de manière à pouvoir les réutiliser facilement pour la description d'autres constructions.

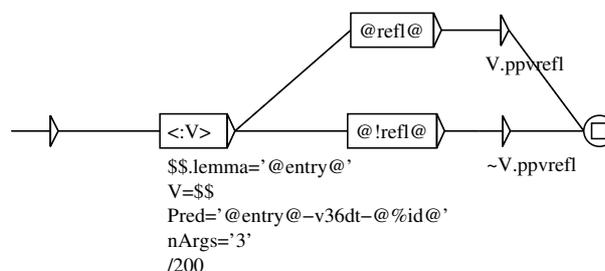


FIG. 4.49 – Sous-graphe V

Le sous-graphe V reconnaît simplement un non-terminal V comme décrit dans la section précédente. L'équation $$$$lemma='@entry@'$ permet de vérifier que le lemme du noyau verbal est bien l'entrée de la table pour laquelle sera générée la grammaire lexicalisée. Les équations $Pred='@entry@-v36dt-@%id@'$ et $nArgs='3'$ servent à positionner les valeurs des traits $Pred$ et $nArgs$ selon les conventions que nous avons vues précédemment (cf. section 4.5.2). Enfin, la propriété $refl$ est une propriété à valeur booléenne que nous avons ajoutée dans les tables que nous traitons et qui spécifie pour chaque entrée, si le verbe est employé avec un pronom réfléchi obligatoire pour cet emploi. Dans ce cas, l'équation $V.ppvrefl$ permet de s'assurer que ce pronom est bien présent dans le noyau verbal reconnu (l'équation $\sim V.ppvrefl$ permet de s'assurer de son absence dans le cas contraire).

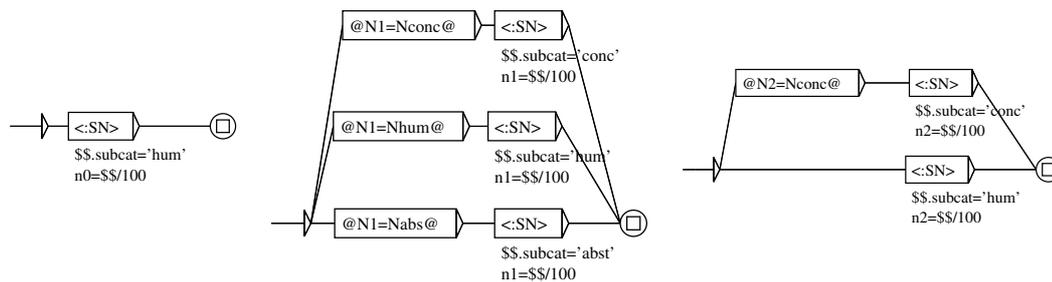


FIG. 4.50 – Sous-graphes N0, N1 et N2

Les sous-graphes N0, N1 et N2 sont présentés dans la figure 4.50. Dans tous les cas, ces arguments sont réalisés sous la forme d’un SN. Les équations sur les traits permettent de préciser la catégorie sémantique du nom tête (nom humain, abstrait ou concret) en fonction des propriétés décrites dans la table. Nous avons pondéré chacune de ces descriptions par un poids de 100 ; en fait, nous avons pondéré ainsi dans toute notre grammaire, l’ensemble des chemins décrivant la réalisation d’un complément essentiel. Cette heuristique nous permet de réduire certaines ambiguïtés d’analyse en préférant les analyses dans lesquelles ont été identifiés le plus grand nombre de prédicats et d’arguments essentiels (par rapport aux analyses où le même complément aurait été identifié comme circonstanciel ou complément du nom par exemple).

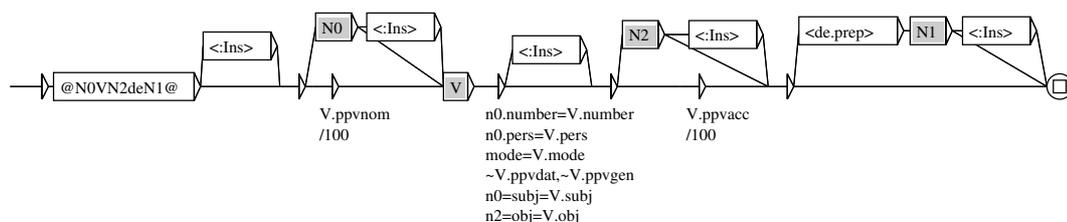


FIG. 4.51 – Sous-graphe NOVN2deN1

Le graphe de la figure 4.51 décrit les constructions associées à la propriété NOVN2deN1 qui indique si le verbe admet la transformation de *croisement des arguments* avec laquelle l’argument N2 passe en position d’objet direct et l’argument N1 est introduit par la préposition *de* :

Max a arnaqué deux cent francs à Luc
 = *Max a arnaqué Luc de deux cent francs*

Ainsi, par exemple, la séquence :

Luc a volé Lea

reçoit deux analyses différentes avec notre grammaire :

`voler-v36dt-271(Luc, Lea, _)`
`voler-v36dt-271(Luc, _, Lea)`

Dans la première, correspondant à la structure de base, *Lea* est l'objet du vol (*Luc a volé Lea à ses parents*) ; dans la seconde, correspondant à la structure croisée, *Lea* est la personne qui a été volée (*Luc a volé Lea de 100 francs*). Cependant, la séquence suivante ne recevra qu'une seule analyse :

Luc a volé un œuf

En effet, le substantif *œuf* est un nom concret et ne peut donc pas être analysé comme l'argument N2 du verbe *voler* qui accepte uniquement un nom humain à cette position.

Enfin, nous présentons, dans la figure 4.52, le sous-graphe P-`prepN` décrivant les phrases avec un complément `miss` extrait et qui est appelé à partir du graphe général décrivant les propositions relatives par exemple (cf. figure 4.40). Nous distinguons deux possibilités : soit le complément extrait n'est pas prépositionnel (et dans ce cas le trait *missprep* a la valeur `zero`), nous reconnaissons alors un phrase privée d'un complément accusatif décrit dans le sous-graphe P-`Nacc` ; soit le complément manquant est un groupe nominal prépositionnel introduit par la préposition *à* (vérifié par l'équation `missprep='à'`), et dans ce cas, la phrase privée d'un tel complément est décrite dans le sous-graphe P-`aN`.

Nous présentons, pour exemple, le graphe P-`Nacc` dans la figure 4.53. Ce graphe reprend la structure du graphe décrivant les formes de base des verbes

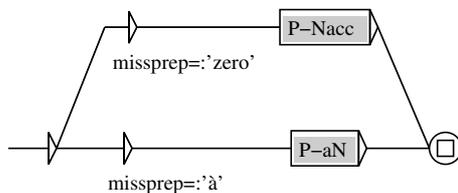


FIG. 4.52 – Sous-graphe P-prepN

de la table (cf. figure 4.48, seulement nous contraignons l’absence de l’argument N1 en surface qui ne peut être réalisé ni sous la forme d’un syntagme nominal (pas d’appel au sous-graphe N1) ni sous la forme d’un pronom clitique (équation $\sim V.ppvacc$). Nous identifions néanmoins cet argument comme le complément extrait avec l’équation $n1=obj=V.obj=miss$. Les équations sur les traits permettent de vérifier que la nature de cet argument manquant est bien compatible avec la nature de l’argument N1 attendu par la prédicat.

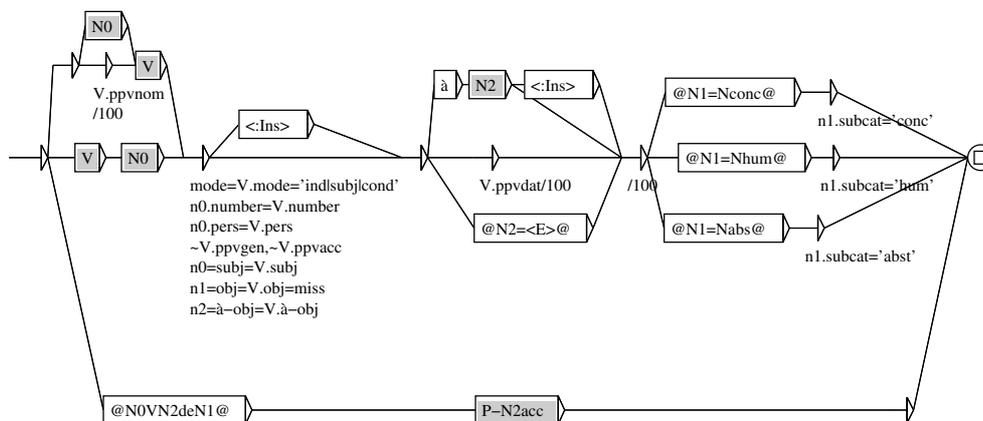


FIG. 4.53 – Sous-graphe P-Nacc

Notre grammaire analyse ainsi les SN suivants :

Lea que Luc a carotté de 100 francs.
Lea que Luc a remboursée

en leur associant les analyses :

```
rembourser-v36dt-215(Luc, _, Lea)
carotter-v36dt-37(Luc, francs, Lea)
```

Nous avons également décrit dans la meta-grammaire de la table les phrases participiales au présent (non terminal **P-ant**) et au passé (**Ppast**), les constructions infinitives (**Pinf**) ainsi que les infinitives privées d'un argument (**Pinf-prepN**). Le graphe **Pinf** est présenté pour exemple dans la figure 4.54. Sa forme reprend celle du graphe représentant les phrases canonique (cf. figure 4.48), excepté pour le sujet qui n'est pas réalisé syntaxiquement dans la proposition infinitive. Nous identifions néanmoins l'argument **N0** du prédicat avec l'équation **n0=subj** ; comme nous le verrons dans la section suivante, la valeur de ce trait **subj** est positionnée depuis le graphe principal d'un verbe à complétive dans le cas où nous pouvons déterminer à quel complément du verbe principal est coréférent le sujet de l'infinitive.

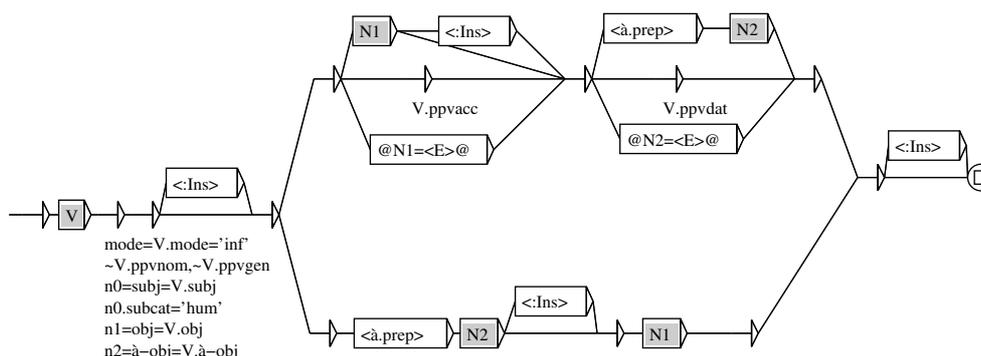


FIG. 4.54 – Sous-graphe **Pinf**

Table 12

La table 12 comporte les verbes à un complément phrastique qui ont la particularité (contrairement aux verbes de la table 6) de rentrer également dans une structure à deux compléments : le premier direct et le second sous

la forme d'une infinitive, introduite par la préposition *de*, dont le sujet est coréférent au premier complément. Par exemple :

Luc approuve que Max soit parti
 = *Luc approuve Max d'être parti*

Nous décrivons dans le graphe de la figure 4.55 ces deux structures définitionnelles.

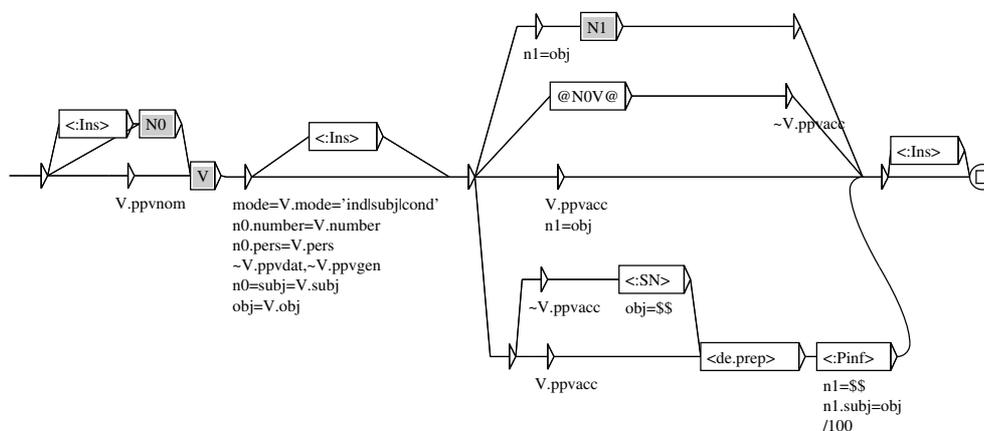


FIG. 4.55 – Formes de base (table 12)

Lorsque nous reconnaissons la construction à deux compléments, nous identifions la proposition infinitive comme l'argument N1 du prédicat (équation $n1=\\$\\$$) et nous rétablissons son sujet, coréférent au complément d'objet direct du verbe principal (équation $n1.subj=obj$). De cette manière, les deux séquences suivantes (considérées comme transformationnellement équivalentes dans le cadre du lexique-grammaire) :

Luc apprécie que Max ait donné une rose à Lea
Luc apprécie Max d'avoir donné une rose à Lea

reçoivent exactement la même analyse en terme d'identification des prédicats et de leurs arguments :

apprécier-v12-9(Luc, donner-v36dt-81(Max, rose, Lea))

A notre connaissance, il n'existe pas d'autre analyseur syntaxique exploitant des ressources linguistiques à large couverture capable de mettre en relation ce type de constructions transformationnellement équivalentes.

Le sous-graphe V (figure 4.56) qui décrit le prédicat verbal est fait sous le même modèle que son homonyme de la table 36DT.

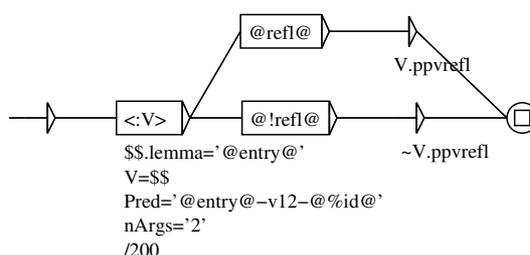


FIG. 4.56 – Sous-graphe V

Les figures 4.57 et 4.58 décrivent les différentes réalisations des arguments N0 et N1. Certains verbes de la table 12 acceptent un sujet non restreint (indiqué par la propriété N0=Nnr) :

(Max+qu'il fasse beau+la fatigue) empêche Luc de travailler

Les autres verbes acceptent un sujet humain uniquement :

*(Max+*qu'il fasse beau+*la fatigue) méprise Luc de travailler*

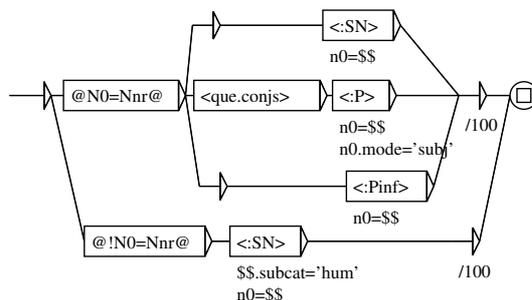


FIG. 4.57 – Sous-graphe N0

L'argument N1, quant à lui, peut être réalisé sous la forme d'une complétive à l'indicatif (propriété $N1=quePind$) ou au subjonctif ($N1=quePsubj$). Pour certains verbes, il est possible de mettre cette complétive à l'infinitif (introduite ou non par la préposition *de*), sans la présence d'un autre complément d'objet direct. Dans ce cas, le sujet de l'infinitive est coréférent au sujet du verbe principal (ce qui est formalisé avec l'équation $n1.subj=n0$) :

*Max adore ($\varepsilon+*de$) faire la vaisselle*
Max arrête ($\varepsilon+de$) faire la vaisselle*
Max exècre ($\varepsilon+de$) faire la vaisselle

Enfin, nous présentons dans la figure 4.59 la meta-grammaire de la table 12 décrivant le non terminal P-prepN. Nous distinguons deux cas :

- soit l'élément manquant est un argument du verbe; Il s'agit alors d'un complément d'objet direct (ce qui est vérifié par l'équation contrainte $missprep= : 'zero'$). Ce complément peut soit être l'argument N1 du prédicat (*La démission de l'ingénieur que Luc critique*) et dans ce cas l'équation $n1=miss$ permet d'identifier l'argument extrait; soit il s'agit du sujet de l'infinitive qui elle est réalisée syntaxiquement dans la construction (*L'ingénieur que Luc critique d'avoir démissioné*), dans ce cas, l'équation $n1.subj=miss$ permet de rétablir cette relation.
- soit l'argument N1 est bien présent dans la construction mais sous la forme d'une proposition subordonnée dans laquelle il manque un élément (l'élément extrait) :

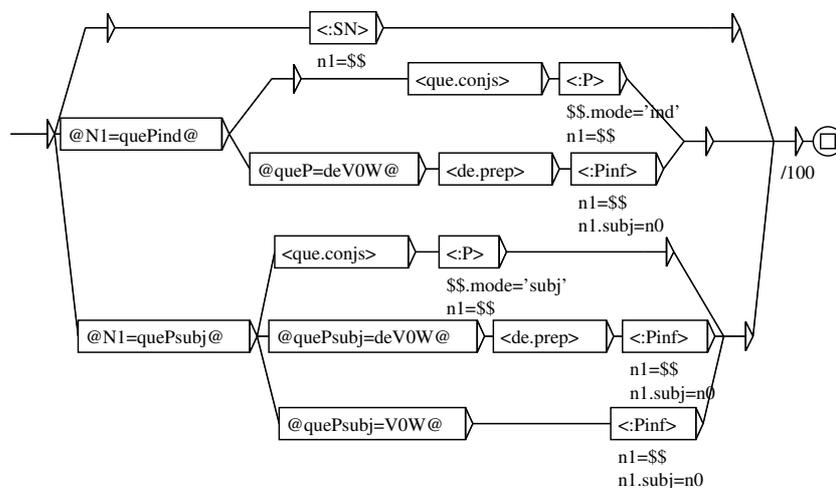


FIG. 4.58 – Sous-graphe N1

Max avec qui Lea apprécierait que Luc travaille
Max avec qui Lea apprécierait de travailler

Dans ces deux phrases, *Lea* est un argument du verbe *Max* et non du verbe *apprécier*. Les équations \hat{miss} et $\hat{missprep}$ permettent de propager les informations sur l'élément extrait au niveau de la proposition enchâssée et ainsi de formaliser ce type de dépendances à distance.

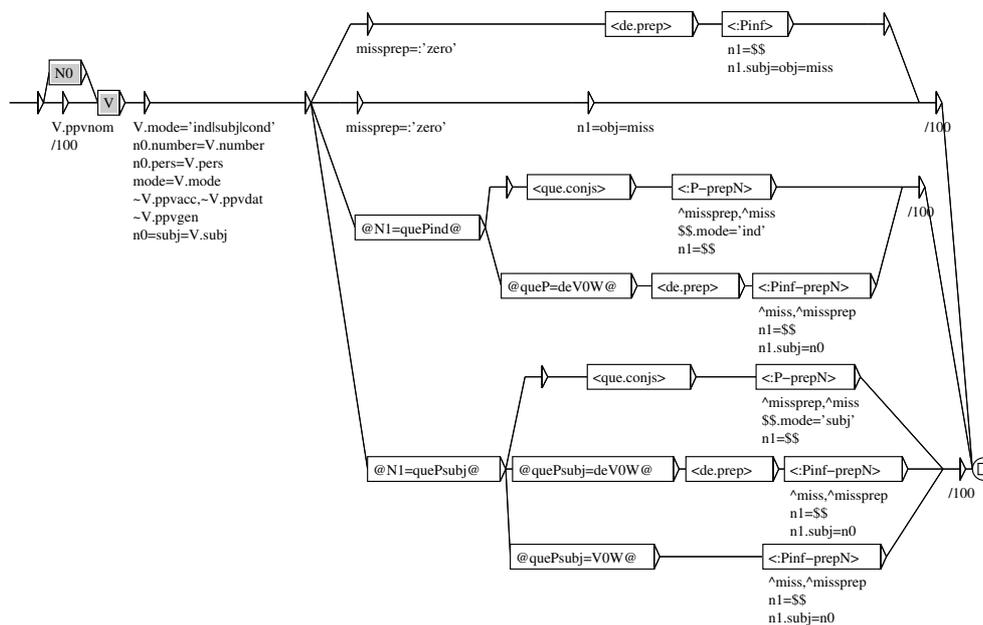


FIG. 4.59 – P-prepN

Table NDR1

Pour terminer cette présentation générale de notre grammaire, nous présentons nos travaux en cours concernant l'intégration des constructions à noms prédicatifs. A cet effet, nous avons entamé la conversion de la table NDR1 de Gaston Gross [Gross, 1989] décrivant les noms prédicatifs qui entrent dans une construction simple à verbe support *donner*, et qui acceptent également une construction converse à verbe support *recevoir*. Le substantif *gifle*, par exemple, appartient à cette table :

Luc a donné une gifle à Lea
 = *Lea a reçu une gifle de Luc*

Nous nous sommes pour l'instant concentré sur la description du premier type de constructions (avec verbe support *donner*) ; le graphe principal décrivant ces formes canoniques est présenté dans la figure 4.60.

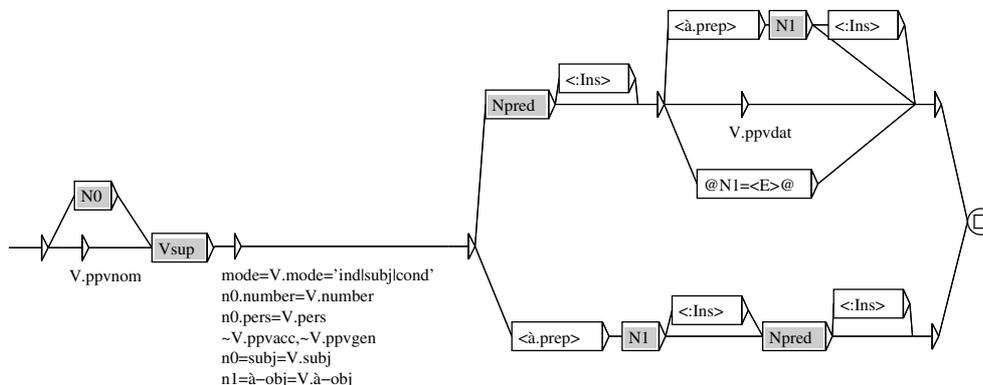


FIG. 4.60 – Constructions canoniques pour la table NDR1

Les graphes de la figure 4.61 présentent les sous-graphes *Vsup*, décrivant la réalisation du verbe support, et le sous-graphe *Npred*, décrivant le nom prédicatif. Nous avons décrit dans le graphe *Vsup*, les possibilités de variations lexicales au niveau du verbe support ; ces variations, qui ont été décrites dans la table, sont sujettes à des contraintes en fonction de l’entrée prédicative :

*Luc a (accordé+apporté+*flanqué) son aide à Lea*
*Luc a (*accordé+*apporté+flanqué) une punition à Lea*

Notre graphe *Npred* qui reconnaît le syntagme nominal à tête prédicative est pour l’instant très simple : il reconnaît simplement un non terminal SN dont le nom tête correspond à l’entrée de la table pour laquelle la grammaire sera lexicalisée (équation $$$.\text{lemma}='@entry@'$). Dans les faits, la forme des groupes nominaux dans lesquels apparaissent ces entrées prédicatives est soumise à certaines restrictions, notamment au niveau du déterminant :

*Luc donne (l’+*une) absolution à Lea*
*Luc donne (*le+un) baiser à Lea*

Ces restrictions étant décrites dans la table, nous pensons, à plus long terme, affiner notre grammaire en remplaçant l’appel au non terminal SN par un

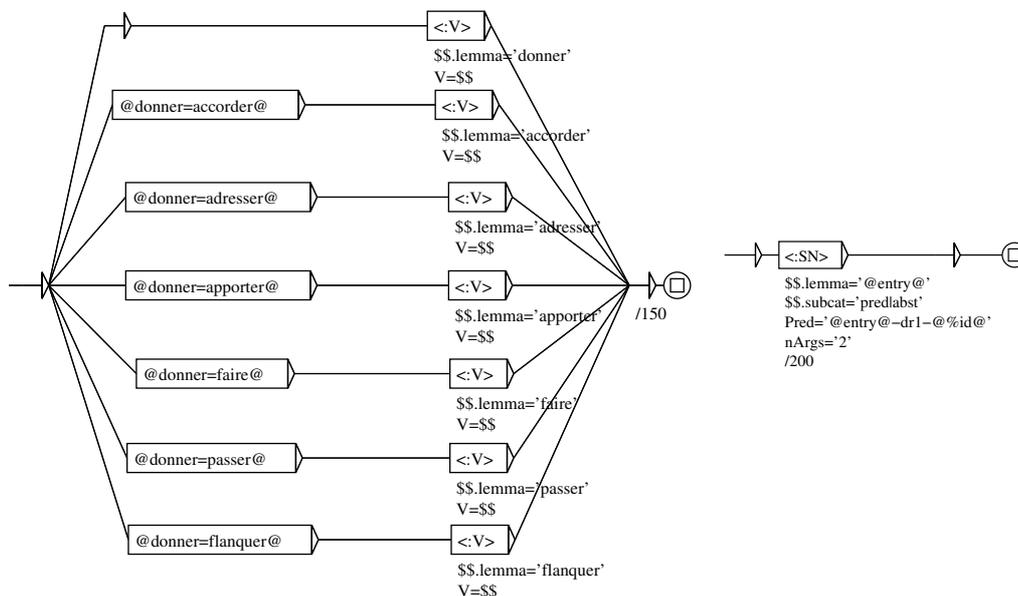


FIG. 4.61 – Sous-graphes Vs et Npred

graphe lexicalisé tenant compte de ces propriétés.

Nous avons pondéré la reconnaissance du verbe support d'un poids de 150 et celle du nom prédicatif d'un poids de 200 (cf. figure 4.61). De cette manière, nous préférons, dans les cas d'ambiguïtés, les analyses correspondant aux constructions à verbe support et nom prédicatif (poids de 350) par rapport aux analyses identifiant la même séquence comme la combinaison d'un verbe plein accompagné d'un de ses compléments (d'un poids $200+100=300$). Ainsi, si nous considérons la séquence suivante :

Luc donne une gifte à Lea

Cette séquence est reconnue par notre grammaire de deux façons différentes : dans la première analyse d'un poids de 500 (200 pour le prédicat + 300 pour les trois arguments), le verbe *donner* dans son emploi décrit dans la table 36DT est identifié comme le prédicat de la phrase et les substantifs *Luc*,

gifle et *Lea* comme ses trois arguments ; dans la seconde analyse, d'un poids de 550 (200 pour le prédicat + 150 pour le verbe support + 200 pour les deux arguments), le substantif *gifle* est le prédicat de la phrase et *Luc* et *Lea* sont ses deux arguments. La seconde analyse ayant un score supérieur, seule celle-ci subsiste à l'issue de l'analyse :

gifle-dr1-128(Luc, Lea)

Les noms prédicatifs n'apparaissent pas nécessairement accompagnés de leur verbe support dans les textes. Ils peuvent être à la tête d'un syntagme nominal dans lequel ses arguments peuvent éventuellement être réalisés sous la forme de compléments du nom :

Max s'étonne que Luc ait flanqué une gifle à Lea
 = *Max s'étonne de la gifle de Luc à Lea [nominalisation]*

Ainsi, de manière à donner aux deux phrases précédentes la même analyse, nous avons commencé à compléter notre grammaire avec la description des SN complexes à noyau prédicatif. Le graphe de ce type de SN pour les entrées de la table NDR1 est présenté, sous sa forme préliminaire, dans la figure 4.62.

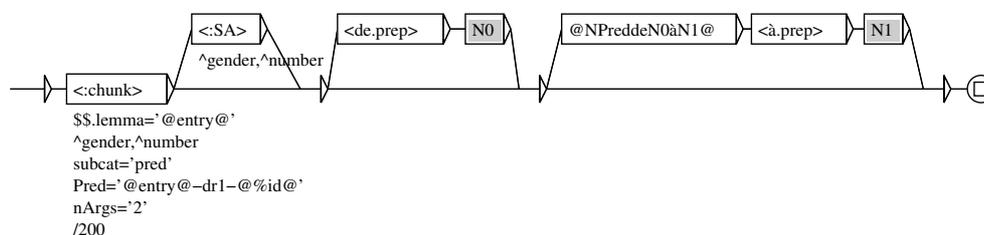


FIG. 4.62 – SN à tête prédicative

Nous décrivons dans ce graphe la possibilité de réaliser l'argument N0 du prédicat sous la forme d'un complément du nom introduit par la préposition *de*. L'argument N1 peut également apparaître au sein du même syntagme nominal, introduit par la préposition *à*, pour les entrées de la table qui acceptent ce type de constructions (propriété *NPredeN0aN1*) :

Luc donne son soutien à Lea
= *le soutien de Luc à Lea*

Cependant, cette description est encore incomplète et même erronée pour certaines entrées. Pour certains noms, c'est l'argument N1 qui est introduit par la préposition *de*, l'argument N0 pouvant éventuellement être introduit par une autre préposition :

Luc accorde son émancipation à Lea
= *l'émancipation de Lea (? par Luc)*

Ces propriétés transformationnelles n'ayant pas été codées dans les tables, nous n'avons pas été en mesure de représenter de manière satisfaisante ce phénomène de nominalisation des phrases à verbe support. Il nous paraît donc nécessaire de compléter la description des tables des noms prédicatifs en étudiant et codant systématiquement ces propriétés manquantes de manière à pouvoir compléter et corriger notre grammaire.

4.6 Résultats

4.6.1 Évaluation sur le corpus TSNLP

Nous avons procédé à une évaluation de notre analyseur et de la couverture de notre grammaire dans son état actuel en l'appliquant sur le corpus TSNLP pour le français [Balkan *et al.*, 1994]. Ce corpus consiste en un jeu de phrases-test représentatif des différentes constructions syntaxiques du français. Nous avons manuellement supprimé de ce corpus les phrases interrogatives et impératives car ces constructions ne sont pas pour l'instant décrites dans notre grammaire. Le corpus résultant est composé de 1 168 phrases. Ce texte a été analysé par notre analyseur en 8 minutes et 20 secondes, ce qui donne une moyenne de 0,42 seconde par phrase. Au total 778 phrases ont reçu une analyse correcte. Nous obtenons donc une couverture de 66% pour l'analyse de ce texte.

La majorité des erreurs d'analyse sont dues à la présence de syntagmes nominaux et adjectivaux complexes à tête prédicative dont le cadre de sous-catégorisation n'est pas décrit dans notre grammaire, comme dans les phrases :

L'ingénieur a obtenu l'accord de l'entreprise pour venir
Jean est affectueux avec l'ingénieur
Jean est heureux que l'ingénieur vienne
Jean est jaloux de l'ingénieur

Toutes ces erreurs n'invalident pas notre méthode : il suffit de décrire le comportement des items lexicaux manquants afin d'élargir la couverture lexicale de notre grammaire et d'analyser ces constructions. Il en est de même pour la description des expressions figées ou locutions verbales, dont on trouve certaines occurrences dans notre corpus de test :

Si l'ingénieur était venu, la réunion aurait eu lieu.

En revanche, d'autres erreurs sont dues à des constructions syntaxiques non décrites dans notre grammaire. La plupart d'entre elles mettent en jeu l'utilisation complexe de conjonctions :

L'ingénieur obtient l'accord et l'homme la permission de l'entreprise pour venir.
L'ingénieur a accepté et signé le contrat

Dans la première phrase, la conjonction *et* coordonne deux phrases qui ont le même verbe en commun (ici, un verbe support) qui est réalisé uniquement dans la première proposition. Pour analyser cette séquence, il faudrait donc mettre en place un mécanisme permettant de rétablir le verbe manquant dans la seconde proposition. Un tel mécanisme peut sans doute être réalisé en propageant les informations sur l'élément manquant dans la seconde proposition à l'aide de contraintes d'unification mais risque d'avoir un impact négatif sur l'efficacité de notre analyseur. La seconde phrase est sans doute encore plus complexe à analyser correctement puisque l'auxiliaire *avoir* modifie les deux verbes au participe passé. De plus, les noms *ingénieur* et *contrat*

sont simultanément arguments des deux prédicats *accepter* et *signer*. Nous n'avons pas pour l'instant trouvé de solution simple pour analyser ce genre de constructions. A notre connaissance, les grammaires formelles du français les plus reconnues pour leur large couverture (telles que FTAG [Abeillé, 2002] ou LFG [Boullier et Sagot, 2005]) ne traitent pas ce type de constructions complexes de manière satisfaisante.

En revanche, nous pensons que notre travail a contribué à faire avancer la recherche en analyse syntaxique sur les deux points suivants, généralement peu ou pas traités dans les différentes approches courantes :

Analyse syntaxique profonde

Beaucoup des analyseurs syntaxiques existants produisent comme résultat un arbre de dérivation représentant la structure syntagmatique de l'énoncé traité (par exemple, la campagne récente EASY d'évaluation des analyseurs syntaxiques du français, demandait aux candidats de produire ce type de résultat). Dans le cadre de notre travail, nous avons tenté de faire de l'analyse syntaxique plus profonde : outre la production d'un arbre de dérivation, nous nous sommes essentiellement intéressé à l'identification des prédicats syntaxiques et de leurs arguments, et ce, indépendamment de leur position ou fonction syntaxique dans l'énoncé. Ce point nous permet notamment de mettre en relation des constructions considérées comme transformationnellement équivalentes, comme les séquences suivantes :

Max apprécie que Luc ait donné une gifle à Lea
Max apprécie la gifle que Luc a donnée à Lea
Max apprécie la gifle de Luc à lea
la gifle de Luc à lea est appréciée (par+de) Max

qui reçoivent exactement la même analyse (en terme de relations prédicat-argument), à l'issue de l'analyse par notre grammaire :

`apprécier-v12-9(Max, gifle-dr1-128(Luc, Lea))`

Distinction des emplois

A l'inverse, un même lemme peut avoir plusieurs sens différents en fonction de la phrase dans laquelle il apparaît. Par exemple, au moins quatre emplois distincts ont été repertoriés pour le verbe *voler* dans le cadre des travaux effectués au LADL :

Max vole chercher Lea (table 2)

Max a volé dans ce livre que Napoleon est mort (table 10)

L'avion vole vers Paris (table 35L)

Max a volé une montre à Lea (table 36DT)

Les distinctions entre ces emplois ayant été faites sur des critères purement formels, nous avons pu les formaliser dans notre grammaire. Nous obtenons donc, à l'issue de l'analyse de ces séquences, des résultats dans lesquels ces emplois ont correctement été distingués :

`voler-v2-107(Max, _, chercher-v32r2-87(Max, Lea))`

`voler-v10-194(Max, mort(Napoleon), dans(livre))`

`voler-v35l-280(avion, vers(Paris))`

`voler-v36dt-271(Max, montre, Lea)`