Implémentation d'une Interface Sémantique-Syntaxe basée sur des Grammaires d'Unification Polarisées

Pierre Lison
plison@agora.eu.org

4 septembre 2006

- Mémoire de fin d'études -

Université Catholique de Louvain
Faculté des Sciences Appliquées
Département d'Ingénierie Informatique

Structure de la présentation

1. Introduction
2. Fondements linguistiques
3. Grammaires d'Unification Polarisées
4. Axiomatisation de GUP
5. Implémentation
6. Validation expérimentale + démonstration
7. Conclusion et Perspectives

1. Introduction

Contexte général du travail réalisé

1. Concerne le Traitement Automatique des Langues Naturelles ;
2. Fondé sur une approche multidisciplinaire (informatique, linguistique, mathématiques) ;
3. Etudie une composante essentielle des modélisations formelles utilisées en TALN : l'interface sémantique-syntaxe ;
4. Aborde cette question dans le cadre d'une modélisation linguistique spécifique - cfr. notamment (2; 4; 3) :
   - *Grammaires d'Unification Sens-Texte* (théorie linguistique)
   - *Grammaires d'Unification Polarisées* (formalisme descriptif) ;

1. Introduction

Trois contributions originales à la recherche en TALN

1. Axiomatisation de GUST/GUP en un problème de *satisfaction de contraintes*, basée sur le récent formalisme grammatical *Extensible Dependency Grammar* [XDG] ;
2. Implémentation de notre interface sur base de cette ”traduction” :
   - Conçeption d'un compilateur de grammaires GUST/GUP $\Rightarrow$ XDG ;
   - Intégration de 8 nouvelles contraintes dans le *XDG Development Kit* [XDK] pour ajuster le programme à nos besoins ;
3. Enfin, validation expérimentale de notre travail par le biais d'un mini-grammaire axée sur le vocabulaire culinaire et une batterie de tests de 50 graphes sémantiques.

2. Fondements linguistiques

Théorie Sens-Texte [TST]

La Théorie Sens-Texte (5) constitue l'inspiration principale du formalisme que nous avons étudié :

1. Théorie linguistique initiée dans les années 60 en ex-URSS ;
2. Se distingue par sa grande richesse et sophistication ;
3. Prend en compte tous les niveaux de la langue ;
4. Le noyau de la structure sémantique est représenté par un graphe de relation prédicat-arguments ;
5. La structure syntaxique est représentée par un arbre de dépendance (6), qui explicite la façon dont les mots, par leur présence, *dépendent* les uns des autres.

2. Fondements linguistiques

Grammaires d'Unification Sens-Texte [GUST]

Les Grammaires d'Unification Sens-Texte (2) - une nouvelle architecture pour la modélisation des langues naturelles :

1. Modèle *mathématique articulé* de la langue ;
2. Synthèse de $\neq$ courants (HPSG, LFG, TAG, et bien sûr la TST) ;
3. Formalisme basé sur l'unification, i.e. la combinaison de structures ;
4. Postule quatre niveaux de représentation :
   - Sémantique (graphe)
   - Syntaxe (arbre de dépendance)
   - Morpho-topologie (arbre ordonné)
   - Phonologie (chaîne linéaire)
5. Grammaire = Règles de *bonne formation* (propre à un niveau donné)
   + Règles d'*interface* entre niveaux ("signes linguistiques").

3. Grammaires d'Unification Polarisées

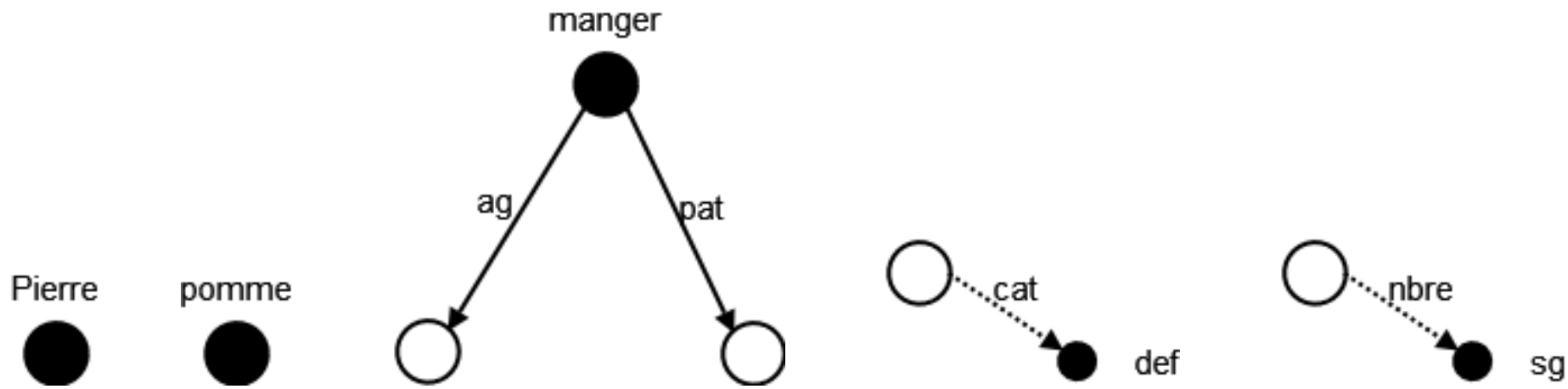
Généralités

Les Grammaires d'Unification Polarisées (3) sont un *formalisme générique* de description linguistique :

1. Formalisme initialement développé pour donner une assise formelle solide à GUST ;
2. Capable de *manipuler* différents types de structures (graphe, arbre, chaîne, etc.) et de les *apparier* ;
3. Contrôle la saturation des structures qu'il combine par une polarisation de leurs objets ;
4. GUP permet de *simuler* élégamment la plupart des formalismes basés sur la combinaison de structures (grammaires de réécriture, de dépendance, LFG, TAG, etc.)

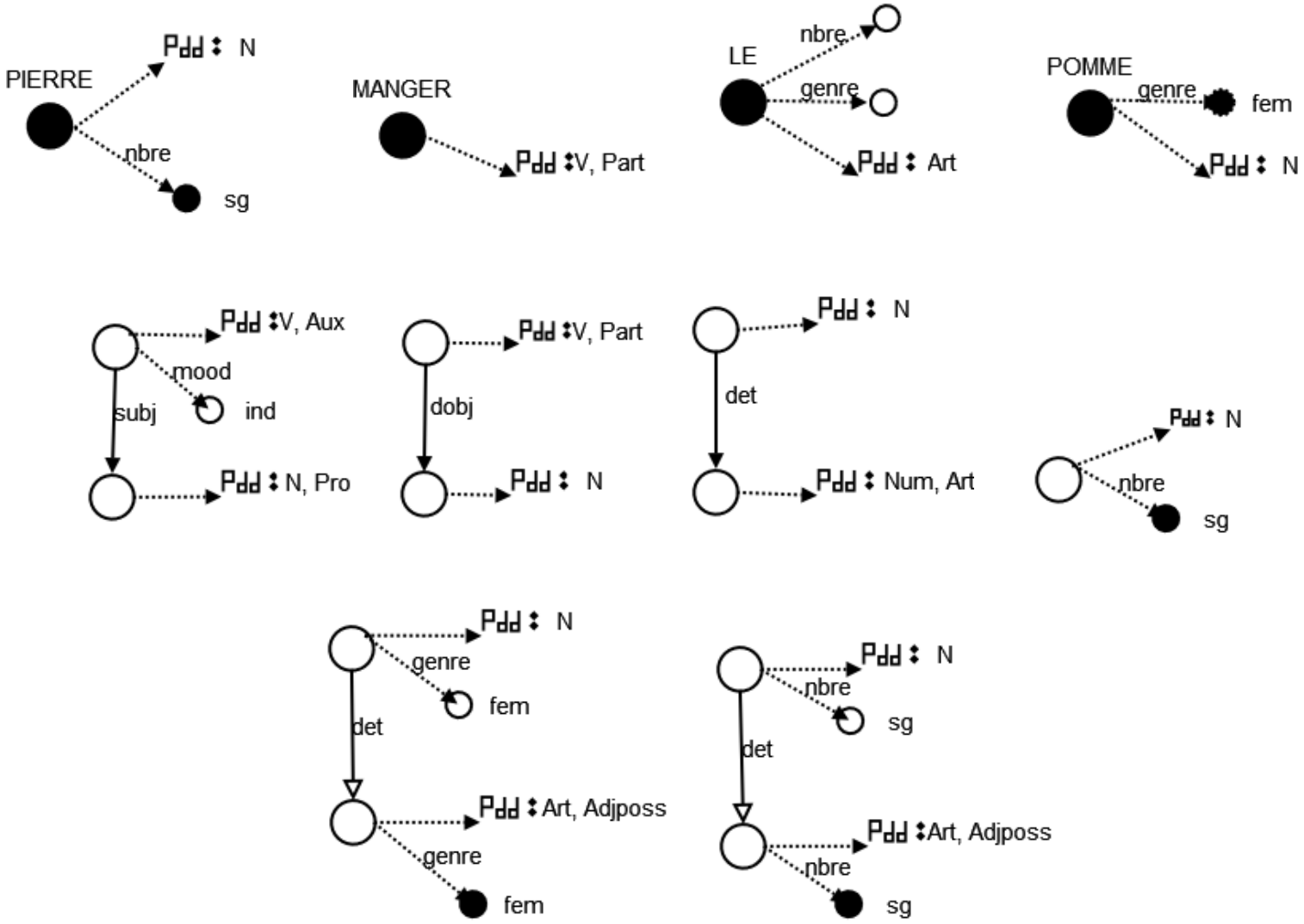
3. Grammaires d'Unification Polarisées

Fragment de la grammaire sémantique de bonne formation $\mathcal{G}_{sem}$



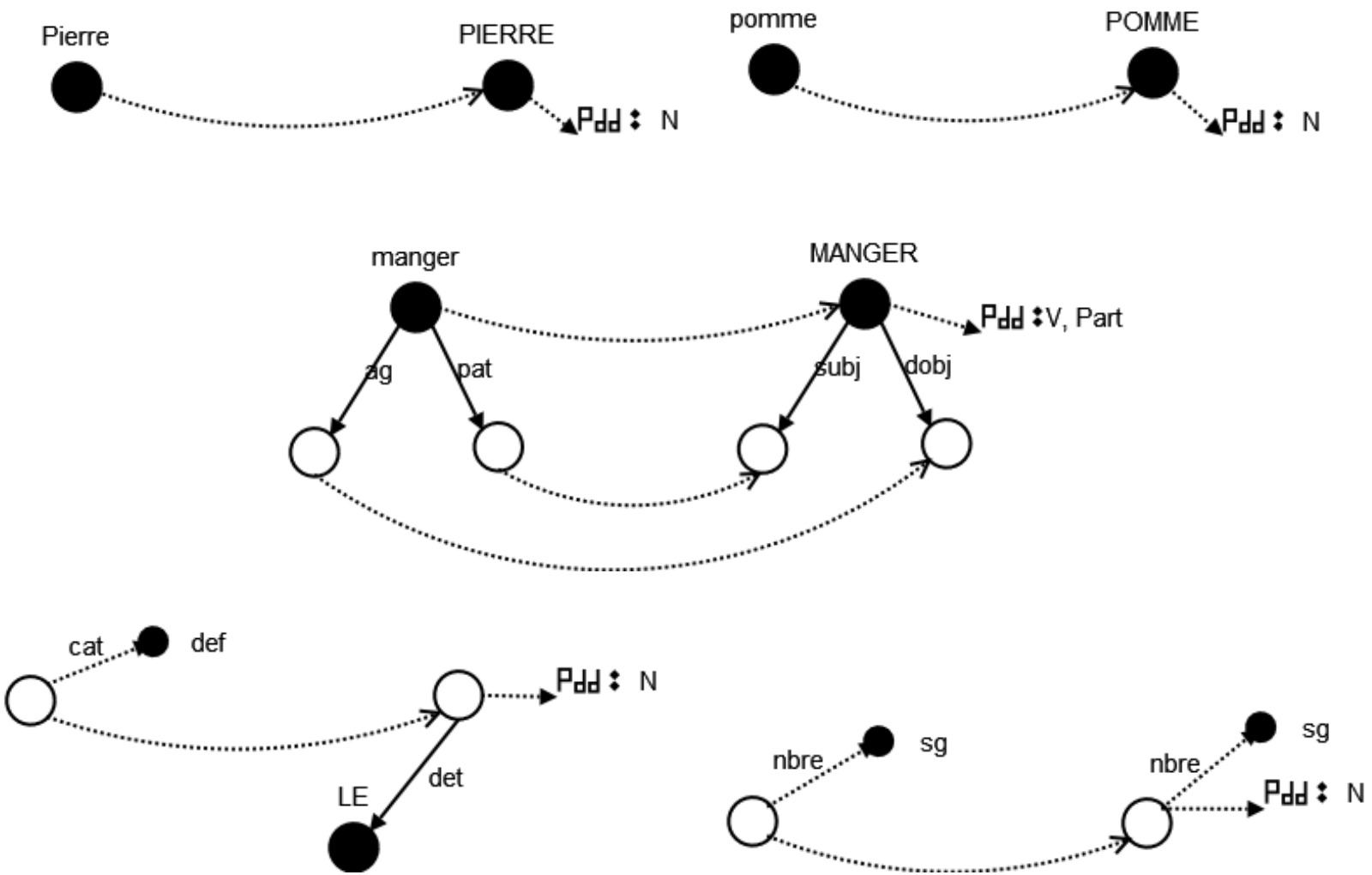
3. Grammaires d'Unification Polarisées

Fragment de la grammaire syntaxique de bonne formation $\mathcal{G}_{synt}$



3. Grammaires d'Unification Polarisées

Fragment de la grammaire d'interface $I_{sem-synt}$



4. Axiomatisation de GUST/GUP

Programmation par contraintes

Nous avons décidé de baser l'implémentation de notre interface sur la programmation par contraintes :

1. Paradigme très intéressant pour le TALN (déclarativité, monotonicité, parallélisme, relativement bonne efficacité) ;
2. L'opération d'analyse / génération est vue comme une *énumération* de modèles bien formés au regard de la grammaire ;
3. *Elimination* progressive des modèles non valables jusqu'à arriver aux solutions (ou à l'insolubilité) ;
4. Deux processus alternant l'un après l'autre :
   - Propagation : application de règles déterministes en vue de réduire l'espace de recherche ;
   - Distribution : choix non déterministe.

4. Axiomatisation de GUST/GUP

Extensible Dependency Grammar [XDG]

XDG (1) est un nouveau formalisme grammatical :

1. Défini comme un langage de description de multigraphes ;
2. Caractéristiques principales :
   - Architecture parallèle ;
   - Utilisation d'arbres de dépendance ;
   - Syntaxe "model-theoretic" ;
   - Basé sur la programmation par contraintes.
3. Existence d'une plateforme de développement de grammaires XDG : *XDG Development Kit*, entièrement conçue sous Mozart/Oz.
4. Nombreux *points communs* avec l'approche sous-tendant GUST/GUP ;
5. ... mais également des *différences* notables, particulièrement concernant la correspondance $M:N$ des unités entre les $\neq$ niveaux.

4. Axiomatisation de GUST/GUP

Traduction de GUST/GUP en XDG

1. Propriété essentielle des GUP : leur monotonicité : pour un système de polarité P = $\{\bigcirc, \textcolor{gray}{\bullet}, \bullet\}$ muni de l'ordre $\bigcirc < \textcolor{gray}{\bullet} < \bullet$, nous avons :

$$\forall x, y \in P, x.y \geq max(x, y) \tag{1}$$

   $\Rightarrow$ Corrolaire de cette propriété : les règles GUP peuvent être appliquées dans n'importe quel ordre !

2. Intuition générale de notre axiomatisation : les contraintes de base de notre grammaire assurent que tous les objets des différents niveaux soient intégralement saturés, i.e.
   - Tous les objets sémantiques ont une polarité $(p_{\mathcal{G}_{sem}}, p_{I_{sem-synt}})$ = ($\bullet$, $\bullet$)
   - Tous les objets syntaxiques ont une polarité $(p_{\mathcal{G}_{synt}}, p_{I_{sem-synt}})$ = ($\bullet$, $\bullet$)

   $\Rightarrow$ 4 contraintes de base à assurer.

4. Axiomatisation de GUST/GUP

Traduction de GUST/GUP en XDG (suite)

3. Pour opérer cette saturation, un ensemble de règles sont spécifiées :
   - Règles sagittales ;
   - Règles d'accord ;
   - Règles d'interface.
4. Ces spécifications font usage d'une série de structures de traits spécialement conçues à cet effet ;
5. Les règles peuvent être associées :
   - à une *unité lexicale* spécifique ;
   - à une *classe* (ensemble d'unités ayant un dénominateur commun)
6. Les règles ne sont opérantes que sous certaines conditions précises et peuvent également ajouter de nouvelles contraintes propres.
7. Lorsque plusieurs saturations distinctes sont possibles pour un même objet, l'on opère une distribution sur ces possibilités.

5. Implémentation

Le logiciel `auGUSTe`

Notre implémentation se compose de :

1. Un compilateur de grammaires GUST/GUP en grammaires XDG :
   - Baptisé `auGUSTe` ;
   - Programmé en Python ;
   - 17 modules, 6.000 lignes de code ;
   - 2 formats d'entrée possibles : graphique (fichier Dia) ou textuel.

2. Un ensemble de 8 principes (= groupe de contraintes) intégré à XDK :
   - Programmé en Mozart/Oz ;
   - Environ 2.000 lignes de code ;
   - Usage de la librairie de contraintes sur les ensembles finis ;
   - Travail d'optimisation de la performance : durée de calcul comprise entre 250 ms et 10 s. pour trouver l'ensemble des solutions ;
   - ... mais ceci reste encore insatisfaisant : divers problèmes encore à résoudre.

6. Validation expérimentale

Méthodologie

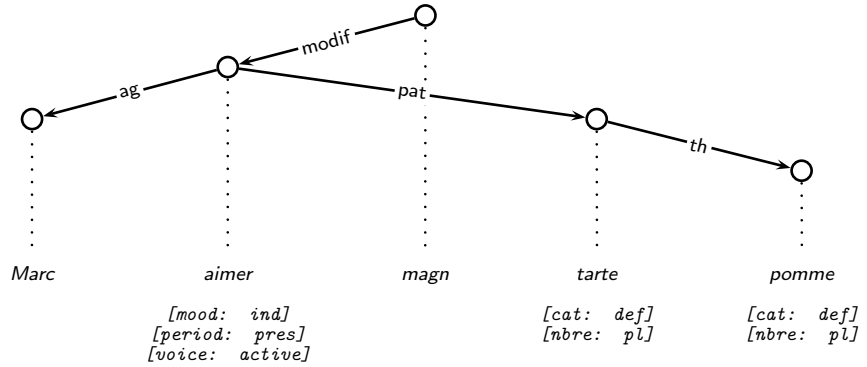
Notre méthode de validation expérimentale :

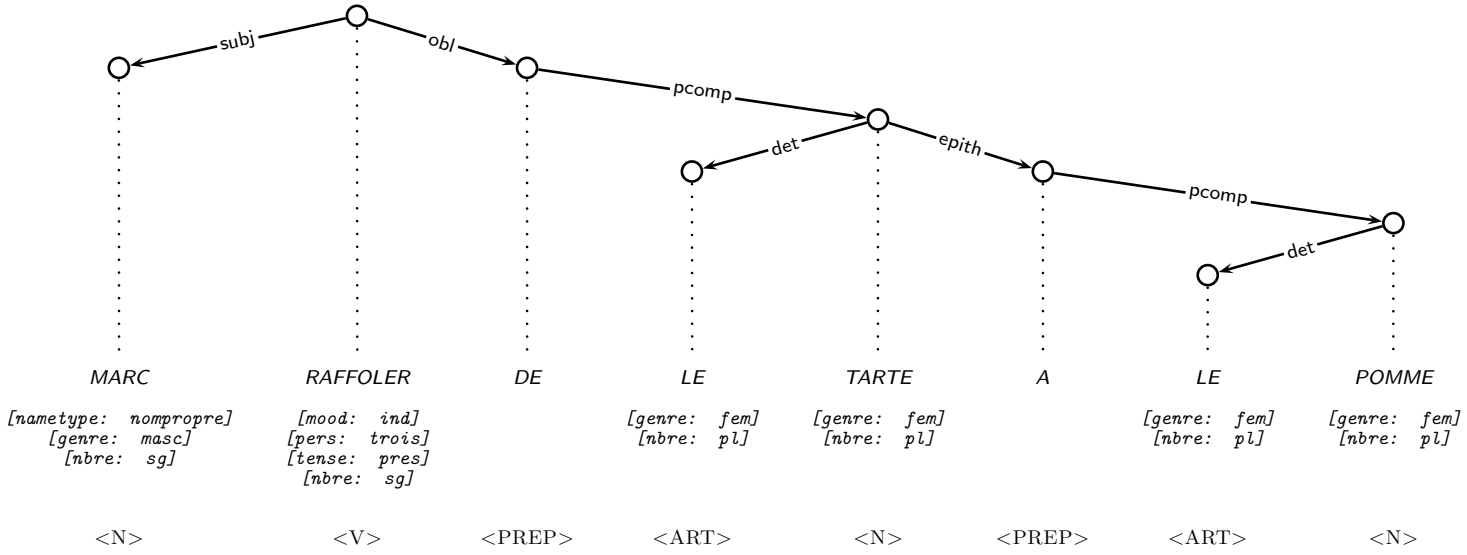
1. Extraction d'une grammaire de base et d'un mini-lexique (quelques centaines de mots) axé sur le vocabulaire culinaire ;
2. Conception via le format graphique (fichiers Dia) d'une mini-grammaire GUST/GUP d'environ 900 règles GUST/GUP ;
3. Vérification de la bonne formation et cohérence interne de notre grammaire ;
4. Création par une personne extérieure d'une batterie de tests de 50 phrases, en veillant à n'utiliser que des mots et constructions autorisés par notre grammaire ;
5. Encodage de la représentation sémantique de ces phrases ;
6. Lancement de l'opération de génération de ces 50 graphes sémantiques, et analyse détaillée des résultats : les problèmes rencontrés proviennent-ils de l'implémentation elle-même ou de la grammaire ?

$\Rightarrow$ Résultats : 48 arbres sur 50 sont correctement générés !

5. Validation expérimentale

Exemple





7. Conclusions et perspectives

1. Sujet particulièrement intéressant :
   - Caractère multidisciplinaire ;
   - A la fois théorique (modélisation) et pratique (implémentation) ;
   - Directement lié à des travaux de recherche en cours.
2. A nécessité une importante quantité de travail :
   - Mise à niveau indispensable en linguistique ;
   - Taille totale de l'implémentation : plus de 8.000 lignes de code ;
   - Constitution d'une véritable mini-grammaire du français.
3. Nombreuses améliorations et extensions possibles :
   - Ajout de dimensions linguistiques supplémentaires ;
   - Modélisations plus poussées (notamment pour l'extraction et la coordination) ;
   - Meilleure formalisation de notre traduction GUST/GUP $\Rightarrow$ XDG ;
   - Et diverses améliorations techniques (performance, robustesse, convivialité).
4. Poursuite du travail actuellement en discussion. Publication ?
5. M'a transmis le goût pour la recherche en TALN !

Références

DEBUSMANN, R. *Extensible Dependency Grammar : A Modular Grammar Formalism Based On Multigraph Description*. PhD thesis, Saarland University, 4 2006.

KAHANE, S. *Grammaire d'Unification Sens-Texte : vers un modèle mathématique articulé de la langue*. Habilitation à diriger des recherches, Université Paris 7, 2002.

KAHANE, S. Grammaires d'unification polarisées. *Actes TALN, Fès* (2004), 233–242.

KAHANE, S., AND LAREAU, F. Grammaire d'unification sens-texte : modularité et polarisation. In *Actes TALN* (Dourdan, 2005), pp. 23–32.

MEL'ČUK, I. Vers une linguistique sens-texte. *Lecon inaugurale au Collège de France. Chaire internationale* (1997).

TESNIÈRE, L. *Eléments de syntaxe structurale*, 2^{e}ẽd. Klincksieck, 1959.