

Évolution vers une grammaire TAG

Eric de la Clergerie et André Bittar

18 Décembre 2008

Introduction

Après avoir exploré l'utilisation des DCG, le but de ce TD est de développer rapidement une grammaire jouet TAG du français.

On rappelle que DyALog permet une structure des grammaires TAG suivant l'architecture XTAG, qui identifie 3 grands niveaux :

Arbres élémentaires Ils sont regroupés en *familles* et peuvent être ancrables (noeud <>).

```
tag_tree{ name=>n0vn1 ,           % Verbe transitif
          family=>n0vn1 ,
          tree => tree s(id=np0 at np , vp(<>v , id=np1 at np))
        }.
```

Lexique syntaxique Il regroupe les lemmes, indiquant pour chaque lemme les familles dont il ancre les arbres. Il est également possible de spécifier des contraintes de co-ancrages et d'équations portant sur les noeuds des arbres ancrés. Un lemme peut ancrer plusieurs familles.

```
tag_lemma(manger , v , [ tag_anchor{ name => [n0vn1 , n0v] } ] ).
tag_lemma(donner , v , tag_anchor{ name => n0vn1pn2 , coanchors => [p=à] } ).
```

Lexique morphologique Il regroupe les formes fléchies, indiquant pour chacune les lemmes associés et les traits morphologiques spécifiques.

```
tag_lexicon(aime , aimer , v , v{ num => sing , pers => 3 } ).
```

Le matériel pour ce TD peut être récupéré sous forme d'archive .tgz à <http://alpage.inria.fr/~clerger/PrologTAL08/td6.tgz>. Il comprend :

- une mini grammaire du français `french.tag`
- une mini-segmenteur `lex2db` permettant de construire un treilli de mots à partir d'une chaîne.
- un fichier `Makefile`

Exercice 1 : Prise en main

Récupérer la grammaire TAG `french.tag`.

Compiler la grammaire, examiner son source et tester quelques phrases comme :

- *Jean mange*
- *Jean mange une pomme*
- *Jean mange une pomme verte*
- *Jean fume*
- * *Jean fume une pomme*

En pratique, commencer par faire

```
make french
```

Les exemples peuvent ensuite être exécutés avec

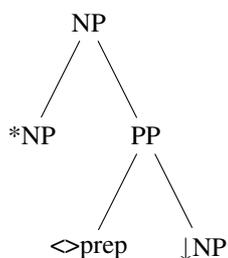
```
> echo "Jean mange" | ./lex2db -word | ./french - -forest
```

Exercice 2 : Gestion des accords

Elle n'est pas faite dans la grammaire de base sauf pour l'arbre `dn` qui gère les groupes nominaux. En vous en inspirant, mettre les conditions d'accord pour les autres arbres (adjectifs et verbes).

Exercice 3 : Groupes prépositionnels

Ajouter une famille `prep` avec les arbres permettant de gérer l'attachement des groupes prépositionnels (**PP**) sur les groupes nominaux ou verbaux.



Tester avec

- *Jean mange la pomme de Pierre*
- *Jean regarde une pomme avec un télescope*
- *Jean regarde une pomme avec le télescope de Pierre*

Exercice 4 : Catégorisation (1)

Ajouter un arbre `n0vn1pn2` et une famille `n0vn1pn2` pour les verbes ditransitifs, comme *donner* :

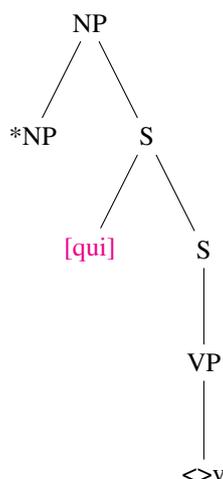
- *Jean donne une pomme à Pierre*

Exercice 5 : Gestion des relatives

Compléter les familles verbales présentes pour inclure la gestion des relatives :

- *Jean regarde Anne qui nage*
- *Jean regarde Anne qui fume un cigare*
- *Jean regarde le cigare que Anne fume*
- *Jean regarde le cigare que Anne donne à Paul*

Note : on pourra s'appuyer sur le modèle suivant :



Extension1 : Gérer les accords entre un antécédent sujet et son verbe :

- **Jean regarde Anne qui nagent*

Extension2 : Avec des modèles similaires, on peut ajouter des arbres pour gérer les formes interrogatives, à condition d'ajouter aux lexiques des entrées pour des pronoms ou déterminants interrogatifs. Ceci permet aussi de traiter les relatives suivantes :

- *Jean regarde Anne à qui Anne donne une pomme*

Exercice 6 : Catégorisation (2)

Ajouter un arbre et une famille v_{VP} pour les verbes sous-catégorisant des infinitives, par exemple pour le verbe *devoir* :

- *Jean doit manger une pomme*
- *Jean peut manger une pomme*
- *Jean doit pouvoir nager*

Extension : Le traitement correct des infinitives oblige à raffiner les familles verbales précédentes pour l'accord de mode entre le verbe et son groupe verbal :

- **Jean doit nage*
- **Jean doit pouvoir nage*
- **Jean doit peut nager*

Exercice 7 : Catégorisation (3)

Ajouter un arbre et une famille v_{VP} pour les verbes sous-catégorisant une complétive, par exemple pour le verbe *penser* :

- *Jean pense que Anne mange une pomme*
- *Jean pense que Anne doit manger une pomme avec Pierre*

Suivant l'exemple du verbe *manger*, assigner une sous-catégorisation multiple au verbe *vouloir* :

- *Jean veut manger une pomme*
- *Jean veut que Anne mange une pomme*
- *Jean doit vouloir manger une pomme*
- *Jean veut devoir nager avec Pierre*

Vérifier que le programme permet l'analyse de la phrase

- *Anne doit penser que Anne doit vouloir regarder la pomme que Paul veut donner à Pierre*

Exercice 8 : Négation

Ajouter les arbres pour traiter les formes simples de la négation :

- *Jean ne mange pas*
- *Jean ne veut pas manger*
- *Jean ne veut pas ne pas manger*
- *Jean ne ne mange pas pas*
- * *Jean veut ne manger pas*

Suggestion : ajouter un trait booléen sur les verbes pour éviter les doubles négations.