Langages hors contexte
Damien Nouvel

Rapport du Projet Argumentation Grand Débat National

Lara DUNUAN et Siyu WANG
Master TAL M1
INALCO

Introduction

Dans le cadre du projet de Grand Débat National de Monsieur Damien Nouvel, ce rapport présente notre travail sur le projet d'argumentation du cours de Langages Hors Contexte.

Le grand débat national a été l'occasion de mettre en place des plateformes contributives, soit à l'initiative du gouvernement sur https://granddebat.fr ou à l'initiative d'un collectif de gilets jaunes sur https://le-vrai-debat.fr. Les données ont été de part et d'autre mises à disposition, elles peuvent donc être téléchargées, puis analysées.

Ce projet demande:

- Partie 1 : un transducteur qui repère des marqueurs d'argumentation (avec Unitex)
- Partie 2 : deux grammaires qui extrait des structures d'arguments :

  * soit avec des automates Unitex (grammaires locales) par inclusion de graphes

  * soit avec le module parsimonious de python

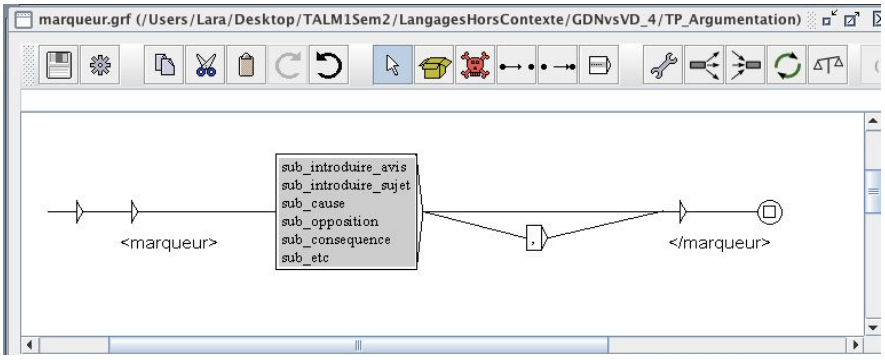
Pour la partie 2, nous avons essayé les 2 méthodes et chaque méthode présente des résultats différents.

Il faut noter que pour la méthode avec python, nous avons aussi écrit des petits programmes de traitement en perl en raison du manque de connaissance sur le module parsimonious, donc il faut exécuter les programmes étape par étape 4 lignes de commande en total pour obtenir le fichier de sortie, vous trouverez les détails dans la partie Méthode Python de ce rapport (lisez moi avant de faire tourner les programmes ;)).

Partie 1

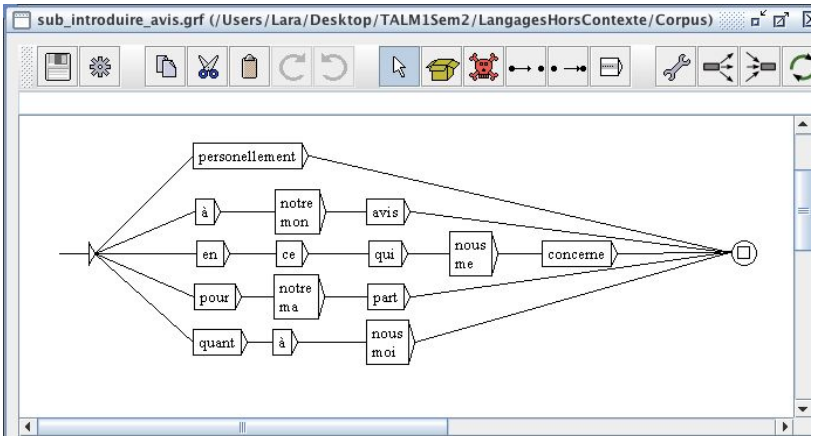
Transducteur avec Unitex

Pour la première partie, nous avons d'abords fait un transducteur qui repère des marqueurs d'argumentation dans notre corpus avec Unitex. Voici une image de notre graphe marqueur.grf:

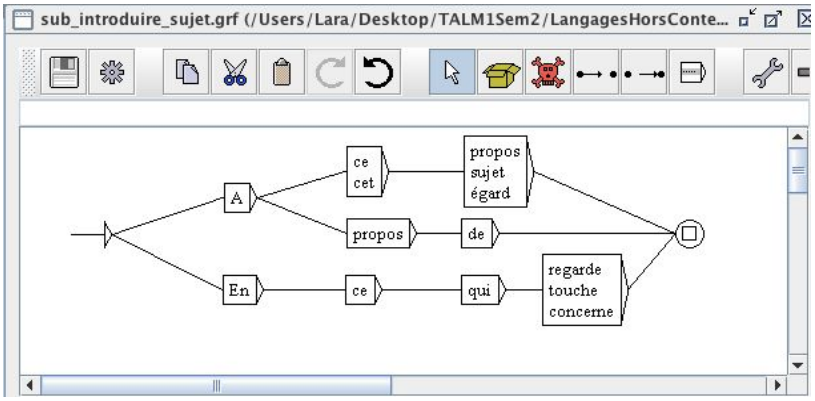


Nous avons créé des sub-graphes pour chaque type de marqueur d'argumentation: pour introduire un avis, pour introduire un sujet, pour exprimer une cause, une opposition et une conséquence, etc. Voici les images des nos sub-graphes:

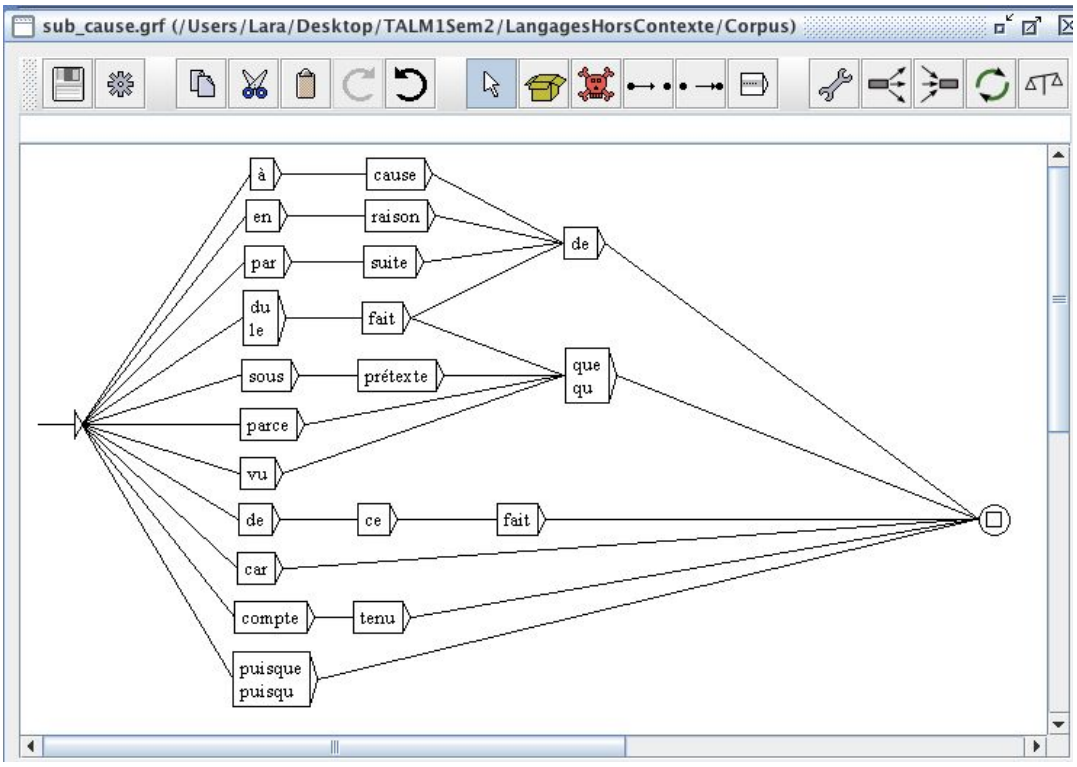
sub_introduire_avis.grf



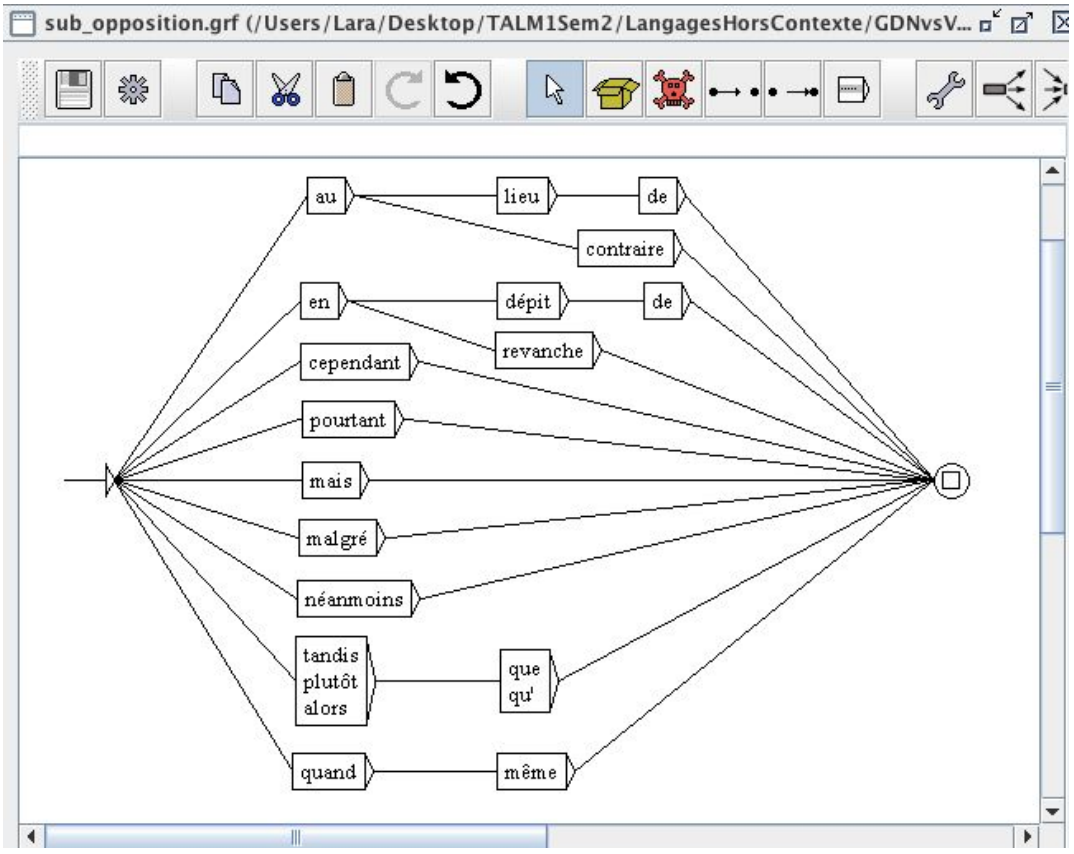
sub_introduire_sujet.grf



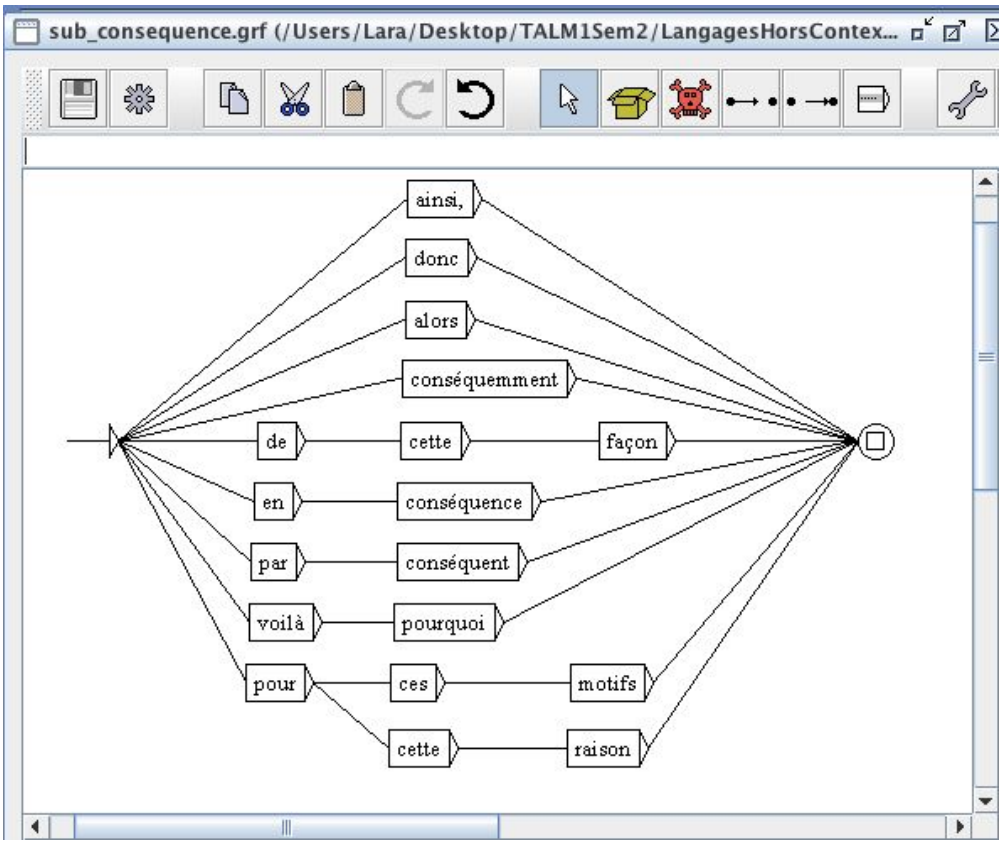
sub_cause.grf



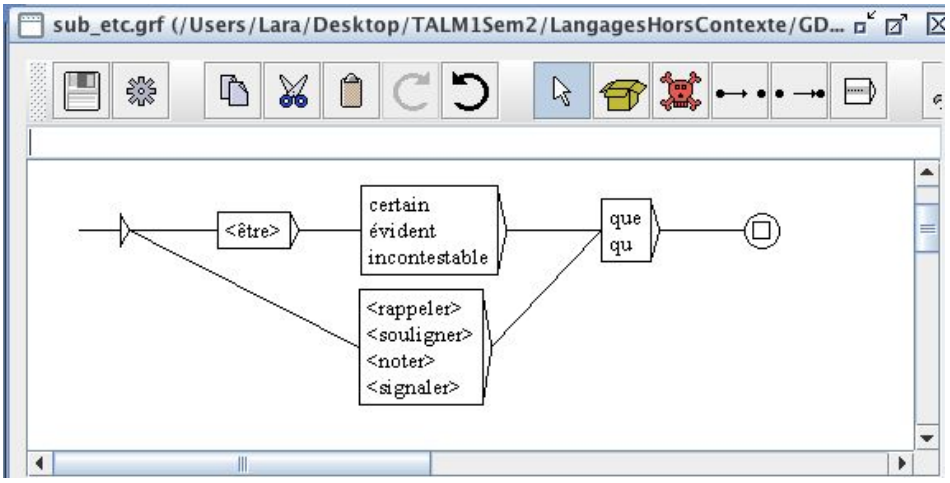
sub_opposition.grf



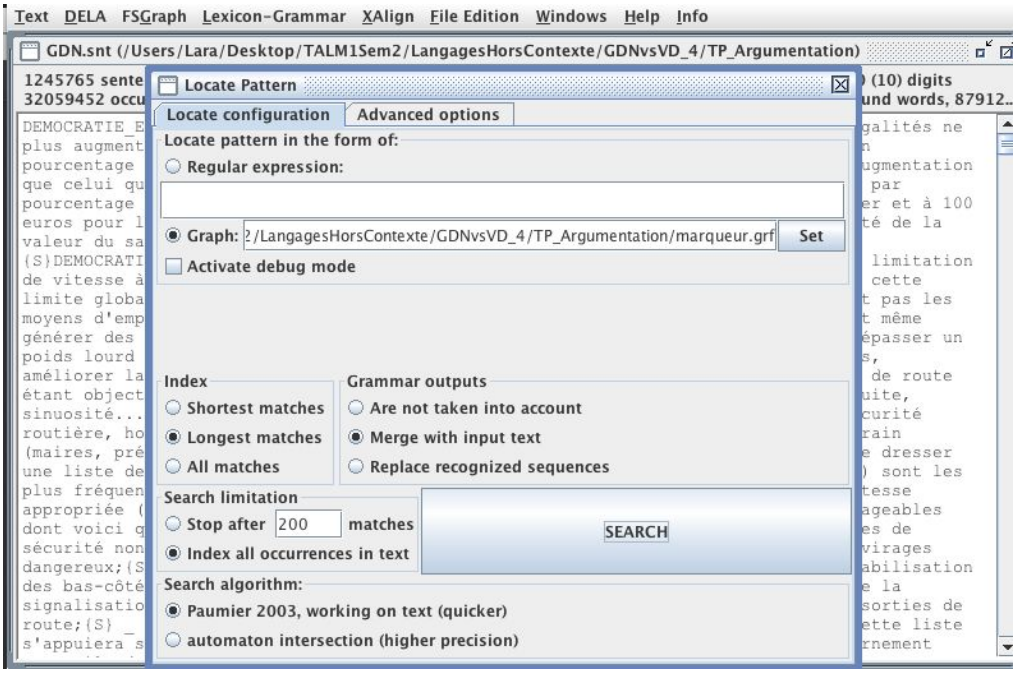
sub_consequence.grf



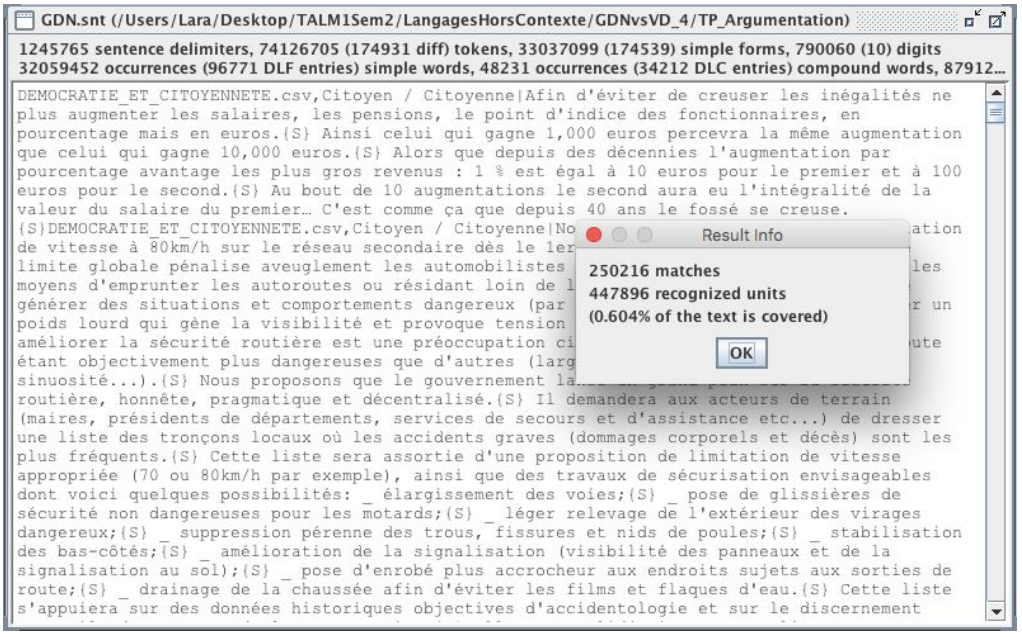
sub_etc.grf



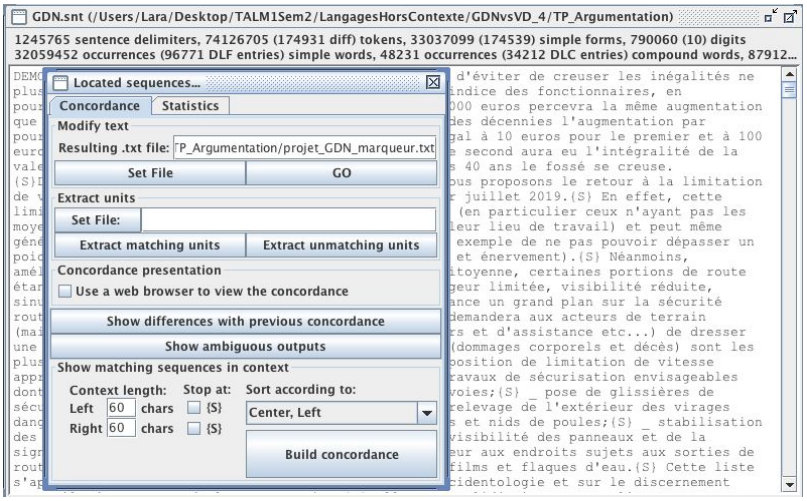
Nous avons appliqué le transducteur à notre corpus (GDN.txt) avec les configurations suivantes:
Index: Longest matches - pour donner la priorité aux séquences les plus longues
Grammar outputs: Merge with input text - permet d'insérer les séquences produites par les sorties.



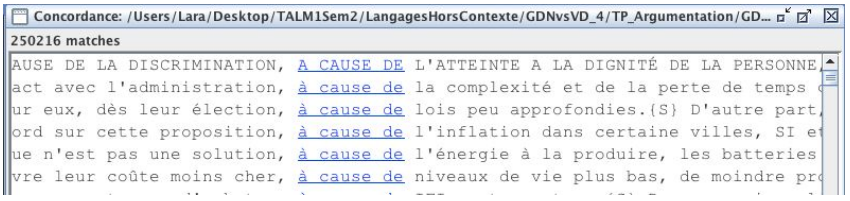
Le résultat de matching se trouve dans l'image suivante:



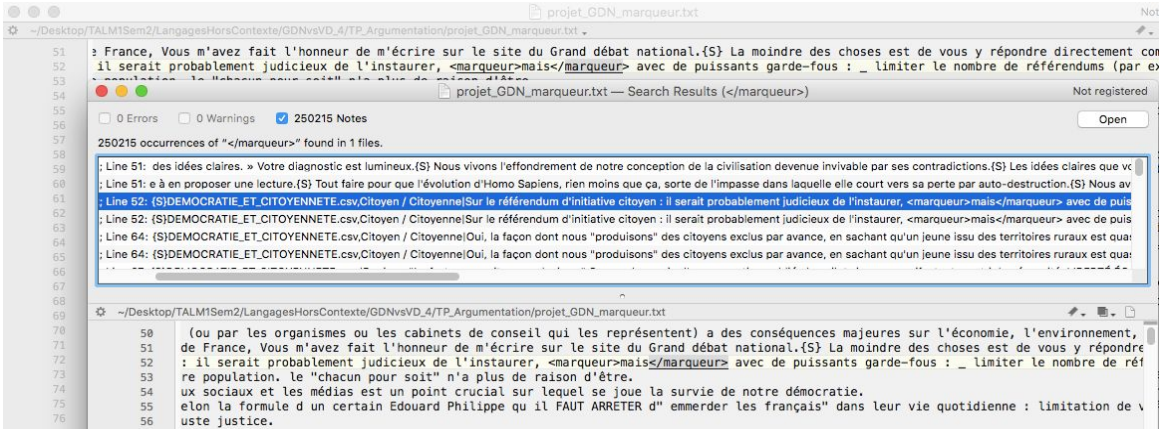
Ensuite, nous avons affiché une concordance avec “Build concordance”:



Voici quelques segments extraits avec notre transducteur:



Dans le cadre “Modify text”, nous avons comme résultat une copie de texte dans laquelle les sorties ont été prises en compte. La concordance complète avec les balises <marqueur> - </marqueur> est sauvegardé dans le fichier projet_GDN_marqueur.txt, voici une illustration:

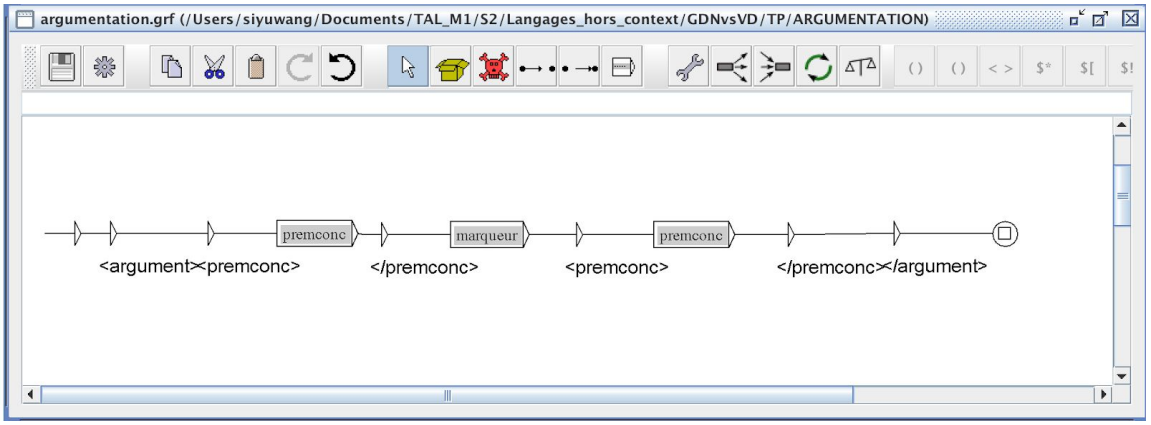


Ce fichier (projet_GDN_marqueur.txt) est le fichier d’entrée pour la deuxième partie: l’extraction des structures d'arguments en utilisant deux grammaires avec le module parsimonious de python.

Partie 2

Méthode 1 : Unitex

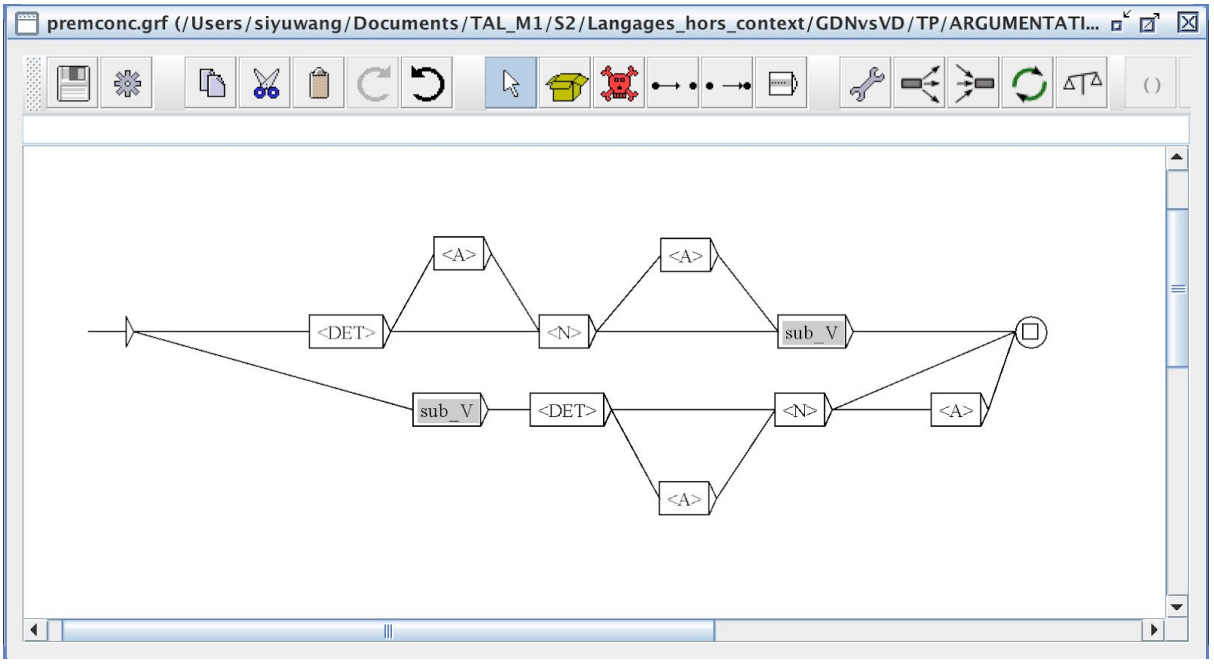
Pour les 2 grammaires, nous avons d'abords essayé de faire des graphes avec l' Unitex pour extraire des segments cibles. Voici une image de notre graphe argumentation.grf



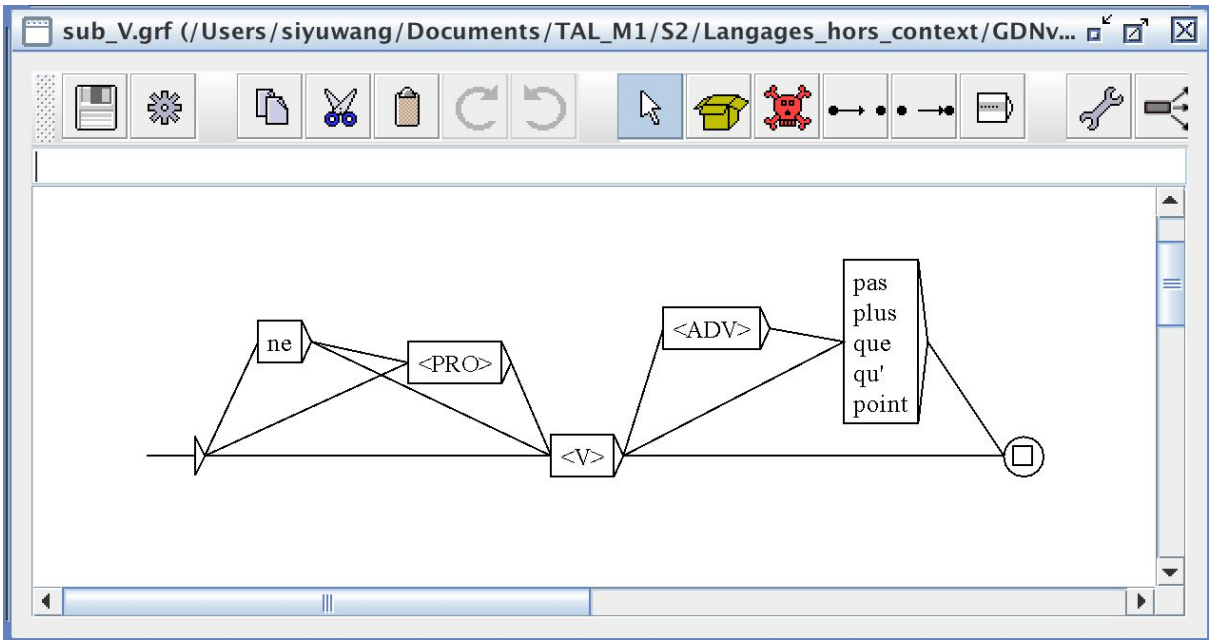
Et comme la position des prémisses et des conclusions varie selon les différents marqueurs, nous avons simplement fait un seul graph qui s'appelle premconc.grf. Ce graphe comprend 2 grammaires :

Première grammaire : DET - ADJ ou non - NOM -ADJ ou non - VERB

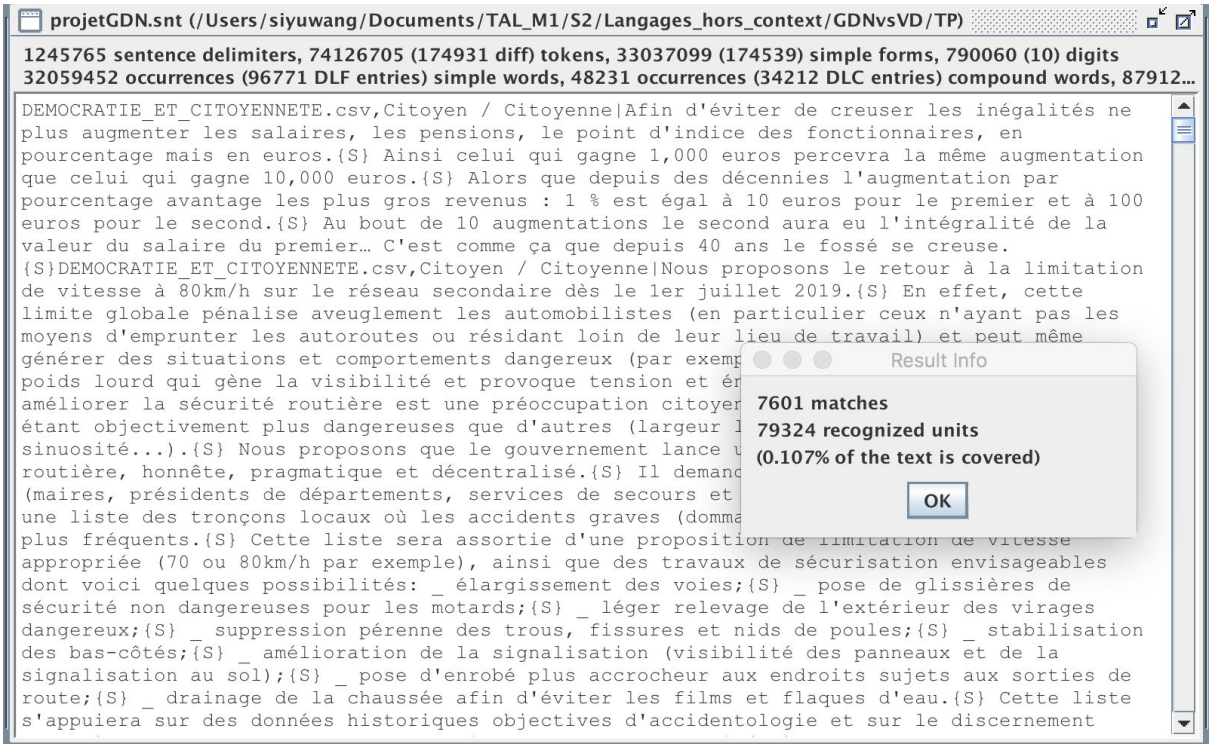
Deuxième grammaire : VERB - DET - ADJ ou nom - NOM - ADJ ou non



Et comme la catégorie VERB comprend également des situations différentes, nous avons aussi fait un sous-graphe des verbes qui s'appelle sub_V.grf.



Le résultat de matching se trouve dans l'image suivante:



Ce procédure a été fait 2 fois, la première sans merging des balises et la deuxième avec le merging des balises.

Voici quelques phrases extraites avec nos 2 grammaires sans merging de balise, et nous pouvons remarquer qu'il y a vraiment de bons résultats.

```
Il n'y a aucune garantie de service alors que l'on parle de services publics

Ne pas lier la transition écologique a des avantages fiscaux car cela crée des bulles spéculatives

les CE devraient participe financièrement a la formation professionnelle plutôt que dépenser des sommes importantes
```

Puis, la concordance complète avec des balises <argument> et <premconc> est sauvegardé dans le fichier concord_unitex.html, voici une illustration:

```
sans concertation et n'<argument><premconc>a aucun sens</premconc><marqueur> alors que</marqueur><premconc> la mortalité routière est</premconc></argument>
globalité, ce chiffre n'<argument><premconc>a aucun sens</premconc><marqueur> car</marqueur><premconc> on mélange les retraites</premconc></argument>
rilles ne correspondant <argument><premconc>a aucun travail</premconc><marqueur> mais</marqueur><premconc> étant le résultat</premconc></argument>
quoi ?{S} Ma remarque n'<argument><premconc>a aucune connotation raciste</premconc><marqueur> car</marqueur><premconc> mon épouse est</premconc></argument>
à se faire.{S} Il n'y <argument><premconc>a aucune garantie de service</premconc><marqueur> alors que</marqueur><premconc> l'on parle de services publics</premconc><
à se faire.{S} Il n'y <argument><premconc>a aucune garantie de service</premconc><marqueur> alors que</marqueur><premconc> l'on parle de services</premconc></argument
primer le Sénat, qui n'<argument><premconc>a aucune utilité</premconc><marqueur> car</marqueur><premconc> le conseil constitutionnel est</premconc></argument>
. ?{S} Ce diplôme forme <argument><premconc>a ce métier</premconc><marqueur> mais</marqueur><premconc> la rémunération ne suit pas</premconc></argument>
. ?{S} Ce diplôme forme <argument><premconc>a ce métier</premconc><marqueur> mais</marqueur><premconc> la rémunération ne suit</premconc></argument>
retraite correspondant <argument><premconc>a ce nouveau travail</premconc><marqueur> alors que</marqueur><premconc> les cotisations retraites sont</premconc></argument
retraite correspondant <argument><premconc>a ce nouveau travail</premconc><marqueur> alors que</marqueur><premconc> les cotisations retraites</premconc></argument>
R JE POURRAI PRETENDRE <argument><premconc>A CETTE ALLOCATION</premconc><marqueur> MAIS</marqueur><premconc> MON BESOIN DU</premconc></argument>
e et fiscale.{S} L'ISF <argument><premconc>a de gros défauts</premconc><marqueur> mais</marqueur><premconc> reste un symbole</premconc></argument>
enne|Aujourd'hui, il y <argument><premconc>a de l'emploi</premconc><marqueur> mais</marqueur><premconc> certains chômeurs gagnent plus</premconc></argument>
enne|Aujourd'hui, il y <argument><premconc>a de l'emploi</premconc><marqueur> mais</marqueur><premconc> certains chômeurs gagnent</premconc></argument>
mmence à réagir s'il y <argument><premconc>a de la casse</premconc><marqueur> alors</marqueur><premconc> il lâche du lest</premconc></argument>
ics et privés.{S} Il y <argument><premconc>a de la place</premconc><marqueur> mais</marqueur><premconc> les propriétaires le refusent</premconc></argument>
abitat ancien?{S} Il y <argument><premconc>a des aides</premconc><marqueur> mais</marqueur><premconc> de nombreuses entreprises profitent</premconc></argument>
NORME OU A FAIRE APPEL <argument><premconc>A DES ASSOCIATIONS</premconc><marqueur> CAR</marqueur><premconc> AUCUNES AIDES NE NOUS AI</premconc></argument>
NORME OU A FAIRE APPEL <argument><premconc>A DES ASSOCIATIONS</premconc><marqueur> CAR</marqueur><premconc> AUCUNES AIDES NE</premconc></argument>
transition écologique <argument><premconc>a des avantages fiscaux</premconc><marqueur> car</marqueur><premconc> cela crée des bulles spéculatives</premconc></argument
transition écologique <argument><premconc>a des avantages fiscaux</premconc><marqueur> car</marqueur><premconc> cela crée des bulles</premconc></argument>
```

Partie 2

Méthode 2 : Python

Fichier de travail: projet_GDN_marqueur.txt (généré par le graph marqueur.grf avec Unitex)

Étape 1: python3 traitement.py

```
(base) MacBook-Pro-de-Siyu:codes siyuwang$ python3 traitement.py
Execution de traitement.py
reading files...
```

Fichier généré : pretraitement.txt

Étape 2: perl clean.pl

```
(base) MacBook-Pro-de-Siyu:codes siyuwang$ perl clean.pl
Execution de clean.pl
```

Fichier généré : out_clean.txt

Étape 3: cat out_clean.txt|bash preproc.sh |python3 gramargs.py > parsim_out.txt

```
(base) MacBook-Pro-de-Siyu:codes siyuwang$ cat out_clean.txt|bash preproc.sh
 |python3 gramargs.py > parsim_out.txt
        reading parameters ...
        tagging ...
10579000         finished.
```

Attention!

Comme c'est un gros corpus, et que nous avons utilisé l'opérateur '>' pour générer le fichier parsim_out.txt, cela prend du temps même si le tagging est marqué "finished", donc il faut patienter ici, cette étape prendra environs 3-5 minutes.

Fichier généré : parsim_out.txt

Étape 4: perl post.pl

```
(base) MacBook-Pro-de-Siyu:codes siyuwang$ perl post.pl
Execution de post.pl
Fichier généré: projet_output.txt
(base) MacBook-Pro-de-Siyu:codes siyuwang$
```

Nous avons voulu comprendre la structure de donnée de l'objet Node généré par le module parsimonious.Grammar, .match(), et .parse(), mais malheureusement nous n'avons pas réussi à écrire l'objet Node dans un fichier txt dans le programme en utilisant la fonction file.write(), parce que cet objet Node ne peut pas être écrit avec .write(). Par conséquent, nous n'avons pas le choix mais de utiliser l'opérateur '>' pour sauvegarder le résultat de parsimonious.

Puis pour ne garder que les segments trouvés qui correspondent à nos 2 grammaires, nous avons ensuite écrit le programme post.pl pour nettoyer le fichier parsim_out.txt. Le fichier final généré est projet_output.txt.

Et dans ce fichier final, les phrases d'arguments extraites sont au format suivant :

<argument><premconc> mots/lemmas/POSs </premconc>
<marqueur>mot/lemma/POS </marqueur> <premconc>mots/lemmas/POSs
</premconc> </argument>