

Analyse syntaxique tabulaire

Cours d'initiation au TAL - M1 SCA & SDL

8 novembre 2013

1 Initiation à l'analyse syntaxique tabulaire

L'analyse syntaxique tabulaire a pour principe de stocker les résultats intermédiaires dans une table (un *chart* en anglais) afin de pouvoir les réutiliser par la suite sans avoir à les recalculer. Ici, les résultats intermédiaires se présentent sous forme d'items $[i, j, w]$ ou $[i, j, C]$, qui signifient que le segment de la phrase analysée entre les positions i et j est un mot w ou un syntagme de catégorie C .

Ouvrez une fenêtre *Idle* et après avoir importé le module *nltk*, exécutez l'instruction `nltk.app.chartparser()`. Une fenêtre, intitulée *Chart Parser Application*, s'ouvre alors, qui va vous permettre de vous initier à l'analyse tabulaire avec une grammaire algébrique.

La grammaire est affichée dans la fenêtre principale *Idle* ainsi que la phrase à analyser : *John ate the cake on the table*.

Au bas de la fenêtre *Chart Parser Application*, vous trouvez différents boutons qui vont vous permettre de mettre en oeuvre différentes stratégies d'analyse. Quelle que soit la stratégie choisie, elle va consister à construire des items de la forme $[i, j, A \rightarrow \alpha \bullet \beta]$. Ces items utilisent la notion de *règle pointée* pour exprimer des syntagmes partiellement analysés. L'item $[i, j, A \rightarrow \alpha \bullet \beta]$ exprime le fait que le segment entre les positions i et j dans la phrase est une suite de syntagmes α et, si l'on peut montrer qu'après la position j il y a une suite de syntagmes β , alors on pourra déduire que l'ensemble forme un seul syntagme de catégorie A .

Pour notre exemple, l'item $[1, 4, VP \rightarrow VP \bullet PP]$ signifie que le segment $[1,4]$, c'est-à-dire *ate the cake* a été analysé comme un groupe verbal (VP) et que si on peut trouver à sa suite un segment $[4, j]$ de catégorie groupe prépositionnel (PP), on pourra analyser le segment $[1,j]$ comme un groupe verbal (VP).

Dans la fenêtre *Chart Parser Application*, les items seront affichés dans le cadre au-dessous de la phrase alors que l'arbre syntaxique apparaîtra dans le cadre au-dessus. Chaque item sera représenté par une flèche couvrant le segment correspondant. Au-dessus de la flèche figure le symbole non terminal partie gauche de la règle pointée, et au-dessous on trouve la partie droite de la règle. La flèche représentant l'item actif est en bleu clair alors que les autres sont en bleu foncé. Un item représentant la phrase complète est en vert : il représente un succès.

Chaque stratégie se définit comme un ensemble de règles de construction d'items. Vous allez utiliser une stratégie *Bottom Up*, c'est-à-dire de type CKY. Cette stratégie

utilise deux règles, *Bottom Up Predict* et *Fundamental*.

La règle *Bottom Up Predict* permet de créer un nouvel item. Elle a deux formes possibles :

- Si w_{i+1} est un mot de la phrase entre les positions i et $i + 1$ et si $A \rightarrow w_{i+1} \alpha$ est une règle de la grammaire, alors on peut créer l’item $[i, i, A \rightarrow \bullet w_{i+1} \alpha]$.
- Si on a établi l’item $[i, j, B \rightarrow \alpha \bullet]$ et si $A \rightarrow B \beta$ est une règle de la grammaire, alors on peut créer l’item $[i, i, A \rightarrow \bullet B \beta]$.

La règle *Fundamental* permet d’avancer dans la réalisation d’une règle. Elle a aussi deux formes possibles :

- Si w_{j+1} est un mot de la phrase entre les positions j et $j + 1$ et si on a établi l’item $[i, j, A \rightarrow \alpha \bullet w_{j+1} \beta]$, alors on peut créer l’item $[i, j + 1, A \rightarrow \alpha w_{j+1} \bullet \beta]$.
- Si on a établi les items $[i, j, A \rightarrow \alpha \bullet B \beta]$ et $[j, k, B \rightarrow \gamma \bullet]$, alors on peut créer l’item $[i, k, A \rightarrow \alpha B \bullet \beta]$.

Les deux boutons *Bottom Up* du bas permettent d’appliquer la stratégie *bottom up* dans sa totalité, c’est-à-dire en combinant les deux types de règles. Les boutons *Bottom Up Predict* et *Fundamental* permettent de séparer l’application des deux types de règles.

Les menus permettent de choisir de procéder pas à pas ou d’un seul coup. Ils permettent aussi de modifier la grammaire et la phrase à analyser. Ils permettent enfin de sauvegarder la grammaire et l’analyse.

Exercice 1.1 Dans la fenêtre de démonstration Chart Parser Application, modifiez la grammaire pour analyser les phrases suivantes :

The big dog is under the table.

John saw the dog and he ate a cake.

John who saw the dog ate a cake.

Analysez chacune des phrase puis après avoir vérifié que les analyses étaient correctes, sauvegardez la grammaire dans un fichier TP4exo1.1.txt

2 Analyse proprement dite

Vous allez maintenant utiliser le *ChartParser*, comme vous avez utilisé le *ShiftReduceParser*. Vous commencerez par créer votre grammaire comme auparavant en la stockant dans une variable *grammar*. à l’aide de l’instruction `nlk.parse_cfg(<texte de la grammaire>)`. Stockez ensuite l’analyseur dans une variable *parser* à l’aide de l’instruction `parser = nlk.ChartParser(grammar, nlk.parse.BU_STRATEGY, trace=2)`. Il fonctionnera ainsi en stratégie *bottom up* et en mode *trace*. Pour obtenir la liste des arbres d’analyse pour une phrase stockée dans une variable *sentence* sous forme d’une liste de mots, vous pourrez utiliser l’instruction `parser.nbest_parse(sentence)`.

Exercice 2.1 Utilisez l’analyseur syntaxique pour analyser les phrases suivantes :

1. Jean mange une pomme et une banane.
2. La pomme sur la table attire Jean.

3. *La pomme que Jean mange est bonne.*
4. *Marie quitte la chambre et Jean entre avec une pomme.*
5. *Jean mange une pomme qui est rouge et bonne.*
6. *Jean vient et mange une pomme.*
7. *Il vient avec une bonne pomme rouge.*

Pour construire la grammaire algébrique nécessaire, vous utiliserez les catégories grammaticales suivantes :

symbole	nom en anglais	nom en français
<i>S</i>	<i>sentence</i>	<i>proposition</i>
<i>NP</i>	<i>noun phrase</i>	<i>syntagme nominal</i>
<i>VP</i>	<i>verb phrase</i>	<i>syntagme verbal</i>
<i>PP</i>	<i>prepositional phrase</i>	<i>syntagme prépositionnel</i>
<i>Conj</i>	<i>coordination conjunction</i>	<i>conjonction de coordination</i>
<i>Det</i>	<i>determiner</i>	<i>déterminant</i>
<i>N</i>	<i>common noun</i>	<i>nom commun</i>
<i>Prep</i>	<i>preposition</i>	<i>préposition</i>
<i>IntrV</i>	<i>intransitive verb</i>	<i>verbe intransitif</i>
<i>TrV</i>	<i>transitive verb</i>	<i>verbe transitif</i>
<i>AttrV</i>	<i>verb with an subject attributive complement</i>	<i>verbe à attribut du sujet</i>
<i>LAdj</i>	<i>left adjective</i>	<i>adjectif épithète gauche</i>
<i>RAdj</i>	<i>right adjective</i>	<i>adjectif épithète droit</i>
<i>RelCl</i>	<i>relative clause</i>	<i>proposition relative</i>
<i>RelPro</i>	<i>relative pronoun</i>	<i>pronon relatif</i>

Vérifiez que l'analyse échoue avec les phrases agrammaticales suivantes :

1. *Jean vient une pomme.*
2. *Jean vient et une pomme.*
3. *La pomme que Jean mange rouge est bonne.*
4. *Jean mange une bonne rouge pomme.*
5. *Il avec une pomme vient.*
6. *Il rouge mange une pomme.*

Mettez le programme dans un fichier TP4exo2.1.py.

3 Envoi des réponses aux exercices

Réunir les différents fichiers contenant les réponses aux exercices dans une archive nommée *<nom-étudiant>.<prénom-étudiant>.TP4.zip* et envoyez-le par mail en pièce jointe à l'adresse *perrier@loria.fr* en mettant dans l'objet du message *initiation au TAL : TP4*.