Modèles de Langage et Analyse Syntaxique

Cours 2 - Syntaxe

Antoine Rozenknop
antoine.rozenknop@lipn.univ-paris13.fr

7 octobre 2010

Plan

Plan

Introduction

Grammaires formelles

- Définitions :
- Hiérarchie des grammaires formelles
  - type 4
  - type 3 : grammaires régulières = grammaires rationnelles
  - type 2 : grammaires hors-contexte = grammaires algébriques
  - type 1 : grammaires contextuelles
  - type 0 : grammaire non-contrainte

Forme Normale de Chomsky

Plan

Plan

Grammaire formelle

- V : Vocabulaire

Grammaire formelle

- V : Vocabulaire
- V_T : Vocabulaire terminal

Grammaire formelle

- V : Vocabulaire
- V_T : Vocabulaire terminal
- V_N : Vocabulaire non-terminlal

Grammaire formelle

- V : Vocabulaire
- V_T : Vocabulaire terminal
- V_N : Vocabulaire non-terminlal
- P : Axiome (élément de V_N)

Grammaire formelle

- V : Vocabulaire
- V_T : Vocabulaire terminal
- V_N : Vocabulaire non-terminlal
- P : Axiome (élément de V_N)
- R : ensemble de règles de récritures de la forme : $\alpha \rightarrow \beta$ avec
  - $\alpha, \beta \in V^*$
  - $\alpha \neq \emptyset$

Grammaire formelle

- V : Vocabulaire
- V_T : Vocabulaire terminal
- V_N : Vocabulaire non-terminlal
- P : Axiome (élément de V_N)
- R : ensemble de règles de récritures de la forme : $\alpha \rightarrow \beta$ avec
  - $\alpha, \beta \in V^*$
  - $\alpha \neq \emptyset$

Définition

$G = (V_N, V_T, R, P)$: grammaire formelle

Dérivation

Définition

Une chaîne u_1 se réécrit en une chaîne u_2 $(u_1 \Rightarrow u_2)$ si et seulement si il existe des chaînes de symboles v_1, v_2, α, β tels que :

1. $u_1 = v_1 \ \alpha \ v_2$
   $u_2 = v_1 \ \beta \ v_2$
2. $\alpha = \beta$ *ou* $(\alpha \rightarrow \beta) \in R$

Dérivation

Définition

Une chaîne u_1 *se réécrit en une chaîne* u_2 $(u_1 \Rightarrow u_2)$ *si et seulement si il existe des chaînes de symboles* v_1, v_2, α, β *tels que :*

1. $u_1 = v_1 \; \alpha \; v_2$
   $u_2 = v_1 \; \beta \; v_2$
2. $\alpha = \beta$ *ou* $(\alpha \rightarrow \beta) \in R$

Définition

Une chaîne u_1 *se dérive en une chaîne* u_2 $(u_1 \Rightarrow^* u_2)$ *si une succession de réécritures permet d'obtenir* u_2 *à partir de* u_1.

Dérivation

Définition

Une chaîne u_1 se réécrit en une chaîne u_2 ($u_1 \Rightarrow u_2$) si et seulement si il existe des chaînes de symboles v_1, v_2, α, β tels que :

1. $u_1 = v_1 \ \alpha \ v_2$
   $u_2 = v_1 \ \beta \ v_2$
2. $\alpha = \beta$ *ou* $(\alpha \rightarrow \beta) \in R$

Définition

Une chaîne u_1 se dérive en une chaîne u_2 ($u_1 \Rightarrow^* u_2$) si une succession de réécritures permet d'obtenir u_2 à partir de u_1.

Définition

Une dérivation est une succession de réécritures.

Langage

Définition

On appelle langage engendré par G l'ensemble de toutes les suites de symboles qui dérivent de l'axiome de G :

$$L(G) = \{x / x \in V^* \text{ et } P \Rightarrow^* x\}$$

Langage

Définition

On appelle langage engendré par G l'ensemble de toutes les suites de symboles qui dérivent de l'axiome de G :

$$L(G)=\{x/x \in V^* \text{ et } P \Rightarrow^* x\}$$

Définition

On appelle langage sur V_T l'ensemble (potentiellement infini) des chaînes de longueurs finies formées avec des éléments du vocabulaire terminal V_T.

Décidabilité

Définition

Un langage est décidable si pour toute phrase on peut savoir au bout d'un temps fini si elle appartient ou nom au langage.

Plan

Grammaires de type 4

Dans une grammaire de type 4, les parties droites de toutes les règles sont des terminaux. Les éléments de R ont donc la forme :

$$X \rightarrow \alpha \text{ avec } \begin{array}{l} \mathrm{X} \in \mathrm{V}_N \\ \alpha \in \mathrm{V}_T{}^* \end{array}$$

Grammaires de type 4

Dans une grammaire de type 4, les parties droites de toutes les règles sont des terminaux. Les éléments de R ont donc la forme :

$$X \to \alpha \text{ avec } \begin{array}{l} \mathrm{X} \in \mathrm{V}_N \\ \alpha \in \mathrm{V}_T{}^* \end{array}$$

Une telle grammaire ne fait qu'énumérer les phrases de son langage sur V_T.

type 3 : grammaires rationnelles

Définition

Dans une grammaire régulière à gauche, les règles ont l'une des formes suivantes :

$$\left.\begin{array}{lll} X & \rightarrow & Y\ a \\ X & \rightarrow & a \end{array}\right\} \text{avec} \left\{\begin{array}{l} X, Y \in V_N \\ a \in V_T \end{array}\right.$$

type 3 : grammaires rationnelles

Définition

Dans une grammaire régulière à gauche, les règles ont l'une des formes suivantes :

$$\left.\begin{array}{lcl} X & \rightarrow & Y\ a \\ X & \rightarrow & a \end{array}\right\} \textit{avec} \left\{\begin{array}{l} X, Y \in V_N \\ a \in V_T \end{array}\right.$$

Définition

Dans une grammaire régulière à droite, les règles ont l'une des formes suivantes :

$$\left.\begin{array}{lcl} X & \rightarrow & a\ Y \\ X & \rightarrow & a \end{array}\right\} \textit{avec} \left\{\begin{array}{l} X, Y \in V_N \\ a \in V_T \end{array}\right.$$

Exemple de grammaire régulière à droite

$$
\begin{array}{lllllll}
P & \rightarrow & a\ P & \qquad & P & \rightarrow & B \\
B & \rightarrow & b & \qquad & B & \rightarrow & b\ B
\end{array}
$$

Exemple de grammaire régulière à droite

$$
\begin{array}{lllllll}
P & \rightarrow & a\ P & \quad & P & \rightarrow & B \\
B & \rightarrow & b & \quad & B & \rightarrow & b\ B
\end{array}
$$

Cette grammaire engendre les chaînes $a^n b^m$.

ab
aab
abb
$aaabbbbbb$
$\ldots$

Utilité en TAL

- la représentation compacte des lexiques

Utilité en TAL

- la représentation compacte des lexiques
- la construction de correcteurs orthographiques ou de lexiques robustes aux erreurs

Utilité en TAL

- la représentation compacte des lexiques
- la construction de correcteurs orthographiques ou de lexiques robustes aux erreurs
- des grammaires locales : mots-composés, séquences acceptables de chiffres (nombres,dates)

Utilité en TAL

- la représentation compacte des lexiques
- la construction de correcteurs orthographiques ou de lexiques robustes aux erreurs
- des grammaires locales : mots-composés, séquences acceptables de chiffres (nombres,dates)
- des grammaires pour des domaines très restreints (annonces de vols dans les aéroports)

Limitations

Les grammaires de type 3 ne peuvent traiter :

- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_1 \ldots b_n$ (“respectivement”)

Limitations

Les grammaires de type 3 ne peuvent traiter :

- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_1 \ldots b_n$ ("respectivement")
- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_n \ldots b_1$ (expressions parenthésées, structures enchâssées) :

En anglais :

The dog the stick the fire burned beat bit the cat

En français :

Le chat que le voisin que le maire que le préfet qui a été condamné a félicité a attrapé est blanc.

Limitations

Les grammaires de type 3 ne peuvent traiter :

- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_1 \ldots b_n$ ("respectivement")
- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_n \ldots b_1$ (expressions parenthésées, structures enchâssées) :
- les chaînes *abac*

  soit ... *soit* ...

  ni ... *ni* ...

Limitations

Les grammaires de type 3 ne peuvent traiter :

- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_1 \ldots b_n$ ("respectivement")
- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_n \ldots b_1$ (expressions parenthésées, structures enchâssées) :
- les chaînes *abac*
- le rejet en fin de phrase des prépositions (en anglais) ou des particules séparables (en allemand) :

  The girl that I do not want to be caught with.

Limitations

Les grammaires de type 3 ne peuvent traiter :

- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_1 \ldots b_n$ ("respectivement")
- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_n \ldots b_1$ (expressions parenthésées, structures enchâssées) :
- les chaînes *abac*
- le rejet en fin de phrase des prépositions (en anglais) ou des particules séparables (en allemand) :
- Les dépendances à longue distance (interrogatives, clivées...)

    Jean veut savoir quelle fille Marie croit que Paul a vue.

type 2 : grammaires hors-contexte

Définition

Une grammaire de type 2, dite hors-contexte ou algébrique, est une grammaire de réécriture dont les parties gauches des règles contiennent un unique non-terminal :

$$X \to \alpha \text{ avec } \begin{cases} X \in V_N \\ \alpha \in V^* \end{cases}$$

Exemple de grammaire hors-contexte

$$
\begin{array}{lll}
P & \rightarrow & a\ P\ b \\
P & \rightarrow & a\ b \\
P & \rightarrow &
\end{array}
$$

Exemple de grammaire hors-contexte

$$
\begin{array}{lll}
P & \rightarrow & a\ P\ b \\
P & \rightarrow & a\ b \\
P & \rightarrow &
\end{array}
$$

Cette grammaire engendre les chaînes $a^n b^n$.

$“”$

$“ab”$

$“aabb”$

$“aaabbb”$

$\ldots$

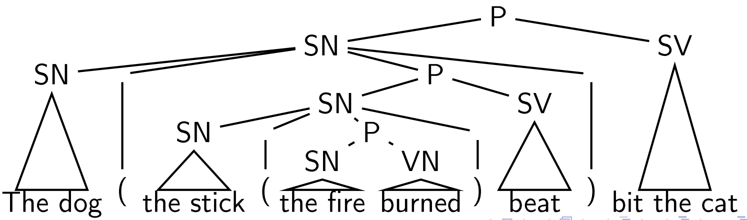
Utilité en TAL

Les grammaires hors-contexte sont adaptées pour :

- les chaînes $a^n b^n$ de type $a_1 \dots a_n b_n \dots b_1$ (expressions parenthésées, structures enchâssées)

$$\begin{array}{rcl} P & \rightarrow & SN\ SV \\ SN & \rightarrow & SN\ (\ P\ ) \end{array}$$ peut analyser :

The dog the stick the fire burned beat bit the cat



Utilité en TAL

Les grammaires hors-contexte sont adaptées pour :

- les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_n \ldots b_1$ (expressions parenthésées, structures enchâssées)
- les chaînes *abac*

$$soit \ldots\ soit \ldots\ (X \rightarrow soit\ Y\ soit\ Y)$$
$$ni \ldots\ ni \ldots\ (X \rightarrow ni\ Y\ ni\ Y)$$

Limitations

- Les grammaires hors-contexte traitent avec difficulté le rejet en fin de phrase des prépositions (en anglais) ou des particules séparables (en allemand).

  Solution : dupliquer les règles pour chaque préposition ou particule

Limitations

- Les grammaires hors-contexte traitent avec difficulté le rejet
- Elles ne peuvent traiter les dépendances à longue distance (interrogatives, clivées...) , le problème étant par exemple de contraindre un accord selon un syntagme qui sort du contexte de cet accord :

  Jean veut savoir quelle fille Marie croit que Paul a vue.

Limitations

- Les grammaires hors-contexte traitent avec difficulté le rejet
- Elles ne peuvent traiter les dépendances à longue distance (interrogatives, clivées. . .)
- Ni les rares langues qui ont des structures de type $a^n b^m c^n d^m$

Limitations

- Les grammaires hors-contexte traitent avec difficulté le rejet
- Elles ne peuvent traiter les dépendances à longue distance (interrogatives, clivées. . .)
- Ni les rares langues qui ont des structures de type $a^n b^m c^n d^m$
- Ni les chaînes $a^n b^n$ de type $a_1 \dots a_n b_1 \dots b_n$ (“respectivement”)

  Henri et Sophie sont repectivement indifférent et séduite par le film.

Limitations

- Les grammaires hors-contexte traitent avec difficulté le rejet
- Elles ne peuvent traiter les dépendances à longue distance (interrogatives, clivées...)
- Ni les rares langues qui ont des structures de type $a^n b^m c^n d^m$
- Ni les chaînes $a^n b^n$ de type $a_1 \ldots a_n b_1 \ldots b_n$ (“respectivement”)

Ces arguments ne sont pas vraiment décisifs pour mettre de côté ces grammaires pour le TAL, car en pratique, les ^{n} et ^{m} ne sont jamais grands.

type 1 : grammaires contextuelles

Définition

Les grammaires de type 1, dites grammaires contextuelles se définissent par des règles du type :

$$\alpha \rightarrow \beta \textit{ avec } \begin{cases} \alpha \in V^+ \\ \beta \in V^* \\ |\beta| \geq |\alpha| \end{cases}$$

(La partie gauche est non vide, et la partie droite doit contenir plus de symboles que la partie gauche)

Exemple :

La grammaire suivante engendre le langage $a^n b^n c^n$

P $\rightarrow$ a B C	C B $\rightarrow$ B C	a B $\rightarrow$ a b
P $\rightarrow$ a P B C		b B $\rightarrow$ b b
		b C $\rightarrow$ b c
		c C $\rightarrow$ c c

Remarques

- Ces grammaires sont décidables

  le nombre de symboles ne peut que croître dans une dérivation.

Remarques

- Ces grammaires sont décidables

  le nombre de symboles ne peut que croître dans une dérivation.

  $\rightarrow$production des phrases de longueur n en temps fini

Remarques

- Ces grammaires sont décidables

  le nombre de symboles ne peut que croître dans une dérivation.

  $\rightarrow$production des phrases de longueur n en temps fini

  $\rightarrow$comparaison de la phrase à analyser avec toutes les productions possibles de longueur n

Remarques

- Ces grammaires sont décidables
- Les grammaires sur lesquelles portent la majorité des recherches en TAL se situent entre le type 1 et le type 2.

Type 0 : grammaire non-contrainte

Définition

Les grammaires de type 0 se définissent par des règles du type :

$$\alpha \rightarrow \beta \textit{ avec } \begin{cases} \alpha \in V^* \\ \beta \in V^* \end{cases}$$

Type 0 : grammaire non-contrainte

Définition

Les grammaires de type 0 se définissent par des règles du type :

$$\alpha \rightarrow \beta \textit{ avec } \begin{cases} \alpha \in V^* \\ \beta \in V^* \end{cases}$$

Remarque : Ces grammaires ne sont pas décidables.

Plan

Forme Normale de Chomsky (CNF)

Définition

Une grammaire hors-contexte est sous Forme Normale de Chomsky si ses règles ont l'une des deux formes :

$$\left.\begin{array}{lcl} X & \rightarrow & Y\ Z \\ X & \rightarrow & a \end{array}\right\} \text{avec} \left\{\begin{array}{l} X \in V_N \\ Y \in V_N \\ Z \in V_N \\ a \in V_T \end{array}\right.$$

Forme Normale de Chomsky (CNF)

Définition

Une grammaire hors-contexte est sous CNF étendue si ses règles peuvent également prendre les formes :

$$\left.\begin{array}{lcl} X & \rightarrow & Y\ Z \\ X & \rightarrow & Y \\ X & \rightarrow & a \end{array}\right\} \; avec \; \left\{\begin{array}{l} X \in V_N \\ Y \in V_N \\ Z \in V_N \\ a \in V_T \end{array}\right.$$

Mise sous forme normale

Pour toute grammaire hors-contexte, il existe une grammaire hors-contexte équivalente sous forme normale.

Mise sous forme normale

Pour toute grammaire hors-contexte, il existe une grammaire hors-contexte équivalente sous forme normale.

Définition

Deux grammaire sont dites équivalentes si elles peuvent produire les mêmes chaînes de symboles terminaux.

Mise sous forme normale

Pour toute grammaire hors-contexte, il existe une grammaire hors-contexte équivalente sous forme normale.

Définition

Deux grammaire sont dites équivalentes si elles peuvent produire les mêmes chaînes de symboles terminaux.

Grammaires de type 2 : passage d'un arbre syntaxique à un arbre équivalent sous CNF = facile

Méthode

3 temps :

1. Suppression des règles de type : $X \rightarrow \alpha \ t_i \ \beta$
2. Suppression des règles de type : $X \rightarrow Y$
3. Suppression des règles de type : $X \rightarrow Y \ Z \ \alpha$

Méthode

3 temps :

1. Suppression des règles de type : $X \rightarrow \alpha\ t_i\ \beta$ (où t_i est un terminal et α et/ou β sont non vides)
   1.1 Créer un non-terminal T_i
   1.2 Ajouter la règle $T_i \rightarrow t_i$
   1.3 Remplacer la règle $X \rightarrow \alpha\ t_i\ \beta$ par $X \rightarrow \alpha\ T_i\ \beta$
2. Suppression des règles de type : $X \rightarrow Y$
3. Suppression des règles de type : $X \rightarrow Y\ Z\ \alpha$

Méthode

3 temps :

1. Suppression des règles de type : $X \rightarrow \alpha\ t_i\ \beta$
2. Suppression des règles de type : $X \rightarrow Y$
3. Suppression des règles de type : $X \rightarrow Y\ Z\ \alpha$

Méthode

3 temps :

1. Suppression des règles de type : $X \rightarrow \alpha\ t_i\ \beta$
2. Suppression des règles de type : $X \rightarrow Y$
   2.1 Pour chaque règle $Z \rightarrow \alpha\ X\ \beta$, ajouter une règle $Z \rightarrow \alpha\ Y\ \beta$.
   2.2 Supprimer $X \rightarrow Y$.
3. Suppression des règles de type : $X \rightarrow Y\ Z\ \alpha$

Méthode

3 temps :

1. Suppression des règles de type : $X \rightarrow \alpha \ t_i \ \beta$
2. Suppression des règles de type : $X \rightarrow Y$
3. Suppression des règles de type : $X \rightarrow Y \ Z \ \alpha$

Méthode

3 temps :

1. Suppression des règles de type : $X \rightarrow \alpha\ t_i\ \beta$
2. Suppression des règles de type : $X \rightarrow Y$
3. Suppression des règles de type : $X \rightarrow Y\ Z\ \alpha$
   3.1 Créer un nouveau non-terminal X_i
   3.2 Ajouter la règle $X_i \rightarrow Z\ \alpha$
   3.3 Remplacer la règle $X \rightarrow Y\ Z\ \alpha$ par $X \rightarrow Y\ X_i$

Méthode

3 temps :

1. Suppression des règles de type : $X \to \alpha\ t_i\ \beta$
2. Suppression des règles de type : $X \to Y$
3. Suppression des règles de type : $X \to Y\ Z\ \alpha$

augmente considérablement le nombre de non-terminaux et de règles.

Exemple :

Forme initiale	Forme normale de Chomsky
R_1 : P $\rightarrow$ SN SV	
R_2 : SN $\rightarrow$ Det N	
R_3 : SN $\rightarrow$ Det N SP	
R_4 : SP $\rightarrow$ Prep SN	
R_5 : SV $\rightarrow$ V	
R_6 : SV $\rightarrow$ V SN	
R_7 : SV $\rightarrow$ V SN SP	
L_5 : V $\rightarrow$ mange	

Exemple :

Forme initiale	Forme normale de Chomsky
R_1 : P $\rightarrow$ SN SV	R_1 : P $\rightarrow$ SN SV
R_2 : SN $\rightarrow$ Det N	R_2 : SN $\rightarrow$ Det N
R_3 : SN $\rightarrow$ Det N SP	
R_4 : SP $\rightarrow$ Prep SN	
R_5 : SV $\rightarrow$ V	
R_6 : SV $\rightarrow$ V SN	
R_7 : SV $\rightarrow$ V SN SP	
L_5 : V $\rightarrow$ mange	

Exemple :

Forme initiale	Forme normale de Chomsky
R_1 : P $\rightarrow$ SN SV	R_1 : P $\rightarrow$ SN SV
R_2 : SN $\rightarrow$ Det N	R_2 : SN $\rightarrow$ Det N
R_3 : SN $\rightarrow$ Det N SP	$R_{3.1}$: X_1 $\rightarrow$ N SP
R_4 : SP $\rightarrow$ Prep SN	
R_5 : SV $\rightarrow$ V	
R_6 : SV $\rightarrow$ V SN	
R_7 : SV $\rightarrow$ V SN SP	
L_5 : V $\rightarrow$ mange	

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN				
R_5 :	SV	$\rightarrow$	V				
R_6 :	SV	$\rightarrow$	V SN				
R_7 :	SV	$\rightarrow$	V SN SP				
L_5 :	V	$\rightarrow$	mange				

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN	R_4 :	SP	$\rightarrow$	Prep SN
R_5 :	SV	$\rightarrow$	V				
R_6 :	SV	$\rightarrow$	V SN				
R_7 :	SV	$\rightarrow$	V SN SP				
L_5 :	V	$\rightarrow$	mange				

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN	R_4 :	SP	$\rightarrow$	Prep SN
R_5 :	SV	$\rightarrow$	V	$R_{1.2}$:	P	$\rightarrow$	SN V
R_6 :	SV	$\rightarrow$	V SN				
R_7 :	SV	$\rightarrow$	V SN SP				
L_5 :	V	$\rightarrow$	mange				

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN	R_4 :	SP	$\rightarrow$	Prep SN
R_5 :	SV	$\rightarrow$	V	$R_{1.2}$:	P	$\rightarrow$	SN V
R_6 :	SV	$\rightarrow$	V SN	R_6 :	SV	$\rightarrow$	V SN
R_7 :	SV	$\rightarrow$	V SN SP				
L_5 :	V	$\rightarrow$	mange				

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN	R_4 :	SP	$\rightarrow$	Prep SN
R_5 :	SV	$\rightarrow$	V	$R_{1.2}$:	P	$\rightarrow$	SN V
R_6 :	SV	$\rightarrow$	V SN	R_6 :	SV	$\rightarrow$	V SN
R_7 :	SV	$\rightarrow$	V SN SP	$R_{7.2}$:	X_2	$\rightarrow$	SN SP
L_5 :	V	$\rightarrow$	mange				

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN	R_4 :	SP	$\rightarrow$	Prep SN
R_5 :	SV	$\rightarrow$	V	$R_{1.2}$:	P	$\rightarrow$	SN V
R_6 :	SV	$\rightarrow$	V SN	R_6 :	SV	$\rightarrow$	V SN
R_7 :	SV	$\rightarrow$	V SN SP	$R_{7.2}$:	X_2	$\rightarrow$	SN SP
				$R_{7.1}$:	SV	$\rightarrow$	V X_2
L_5 :	V	$\rightarrow$	mange				

Exemple :

Forme initiale				Forme normale de Chomsky			
R_1 :	P	$\rightarrow$	SN SV	R_1 :	P	$\rightarrow$	SN SV
R_2 :	SN	$\rightarrow$	Det N	R_2 :	SN	$\rightarrow$	Det N
R_3 :	SN	$\rightarrow$	Det N SP	$R_{3.1}$:	X_1	$\rightarrow$	N SP
				$R_{3.2}$:	SN	$\rightarrow$	Det X_1
R_4 :	SP	$\rightarrow$	Prep SN	R_4 :	SP	$\rightarrow$	Prep SN
R_5 :	SV	$\rightarrow$	V	$R_{1.2}$:	P	$\rightarrow$	SN V
R_6 :	SV	$\rightarrow$	V SN	R_6 :	SV	$\rightarrow$	V SN
R_7 :	SV	$\rightarrow$	V SN SP	$R_{7.2}$:	X_2	$\rightarrow$	SN SP
				$R_{7.1}$:	SV	$\rightarrow$	V X_2
L_5 :	V	$\rightarrow$	mange	L_5 :	V	$\rightarrow$	mange