A Parser for Pregroup Grammars Based on Partial Composition

Denis Béchet

LINA, Université of Nantes

Pregroup

- Pregroup : *ordered monoid* $(P, \leq, ., 1)$ where every a has a left (right) adjoint a^l (a^r) such that: $a^l a \leq 1 \leq a a^l$ ($a a^r \leq 1 \leq a^r a$) and $a \leq b \Longrightarrow cad \leq cbd$
- Properties:
  $a^{rl} = a = a^{lr}$, $1^r = 1 = 1^l$ $(a.b)^r = b^r.a^r$, $(a.b)^l = b^l.a^l$
- Iterated Adjoints :
  $a^{(0)} = a$, $a^{(i-1)} = (a^{(i)})^l$, $a^{(i+1)} = (a^{(i)})^r$
- Free Pregroup based on $(Pr, \leq_{Pr})$:
  - Atomic types : $(Pr, \leq_{Pr})$
  - Simple types : $a^{(i)}$
  - Types : $a_1^{(i_1)} \cdots a_k^{(i_k)}$

LINA LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE — UNIVERSITÉ DE NANTES

Pregroup Grammar

- $Type$: list of simple types
- Grammar : $G \subset \Sigma \times Type$ (Σ words, G finite)
- $L(G) \subseteq \Sigma^+$: set of sequence of words such that the concatenation of types is $\leq s$
- Example : Let $P = \{\pi_1, \pi_2, \pi_3, s, s_1, s_2, p_1, p_2, o, ...\}$ with
  - π_i : subject (I : π_1, you, we, they : π_2, he, she, it : π_3)
  - o : direct object
  - n : name or noun phrase ($n \leq \pi_3$ and $n \leq o$)
  - s_i : present (i=1) or past (i=2) sentence
  - s : correct sentence ($s_1 \leq s$ and $s_2 \leq s$)
  - p_i : present (i=1) or past (i=2) participle

Examples

he loves her

$\pi_3 \quad \pi_3^1 \underline{s_1} o^{-1} \quad o$

a link between a *and* b
means : $ab \leq 1$

John loves Mary

$n \quad \pi_3^1 \underline{s_1} o^{-1} \quad n$

because : $n \leq \pi_3 \Longrightarrow n\pi_3^1 \leq \pi_3\pi_3^1 \leq 1$
and : $n \leq o \Longrightarrow o^{-1}n \leq o^{-1}o \leq 1$

you have been seing her

$\pi_2 \quad \pi_2^1 \underline{s_1} p_2^{-1} \quad p_2 p_1^{-1} \quad p_1 o^{-1} \quad o$

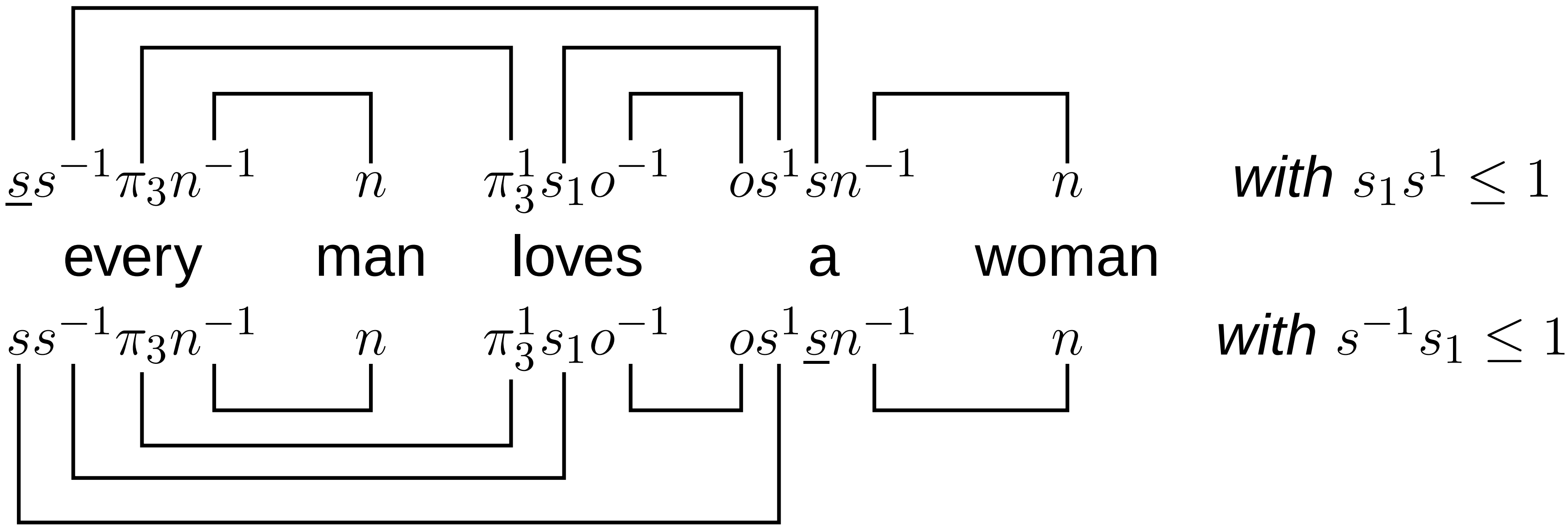
Examples

I	have	said	I	saw	her	*because :*
π_1	$\pi_1^1 \underline{s_1} p_2^{-1}$	$p_2 s^{-1}$	π_1	$\pi_1^1 s_2 o^{-1}$	o	$s_2 \leq s \Longrightarrow s^{-1} s_2 \leq 1$

every	man	runs	*because :*
$\underline{s} s^{-1} \pi_3 c^{-1}$	c	$\pi_3^1 s_1$	$s_1 \leq s \Longrightarrow s^{-1} s_1 \leq 1$

John	loves	every	woman	*because :*
n	$\pi_3^1 s_1 o^{-1}$	$o s^1 \underline{s} c^{-1}$	c	$s_1 \leq s \Longrightarrow s_1 s^1 \leq 1$

A More Complex Example



Parsing using Partial Composition

Parsing of "whom have you seen ?" ($q' \leq_{Pr} s$)

whom	have	you	seen
$q'o^{ll}q^l$	$qp_2^l\pi_2^l$	π_2	p_2o^l

$$\boxed{\boxed{q'o^{ll}q^l,\ \boxed{qp_2^l\pi_2^l, \pi_2}^{[1]}}^{[1]}\quad , p_2o^l}^{[2]}$$

Parsing Using GCON

s an atomic type : $v_1 \cdots v_n \in \mathcal{L}(G)$ iff
for $1 \leq i \leq n$, $\exists X_i \in I(v_i)$ and $\exists s' \in Pr$ such that :

$$\begin{cases} X_1 \cdots X_n \stackrel{(GCON)^*}{\longrightarrow} s' \\ s' \leq_{Pr} s \end{cases}$$

$\stackrel{(GCON)^*}{\longrightarrow}$: reflexive and transitive closer of $\stackrel{(GCON)}{\longrightarrow}$:

$$Xp^{(n)}q^{(n+1)}Y \stackrel{(GCON)}{\longrightarrow} XY$$

if $p \leq_{Pr} q$ and n is even or if $q \leq_{Pr} p$ and n is odd

LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE
UNIVERSITÉ DE NANTES

Example

Parsing of “whom have you seen ?” ($q' \leq_{Pr} s$)

whom	have	you	seen
$q' o^{ll} q^l$	$q p_2^l \pi_2^l$	π_2	$p_2 o^l$

$$q' o^{ll} q^l q p_2^l \pi_2^l \pi_2 p_2 o^l \xrightarrow{(GCON)} q' o^{ll} p_2^l \pi_2^l \pi_2 p_2 o^l$$

$$\xrightarrow{(GCON)} q' o^{ll} p_2^l p_2 o^l$$

$$\xrightarrow{(GCON)} q' o^{ll} o^l$$

$$\xrightarrow{(GCON)} q'$$

and $q' \leq_{Pr} s$

LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE

UNIVERSITÉ DE NANTES

“Pregroup Net”

Parsing of “whom have you seen ?” ($q' \leq_{Pr} s$)

whom	have	you	seen
$q'o^{ll}q^l$	$qp_2^l\pi_2^l$	π_2	p_2o^l

$$q'o^{ll}q^lqp_2^l\pi_2^l\pi_2p_2o^l \qquad (q' \leq_{Pr} s)$$

Partial Command

- $[\mathrm{C}]$ **(partial composition)** : for $k \in \mathbb{N}$,

$$\Gamma, X p_1^{(n_1)} \cdots p_k^{(n_k)}, q_k^{(n_k+1)} \cdots q_1^{(n_1+1)} Y, \Delta \xrightarrow{E} \Gamma, XY, \Delta$$

if $p_i \leq_{Pr} q_i$ and n_i is even or if $q_i \leq_{Pr} p_i$ and n_i is odd, for $1 \leq i \leq k$.

LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE
UNIVERSITÉ DE NANTES

Partial Composition (Example)

$$\boxed{q' o^{ll} q^l, q p_2^l \pi_2^l}^{[1]}, \pi_2, p_2 o^l \xrightarrow{C} \boxed{q' o^{ll} p_2^l \pi_2^l, \pi_2}^{[1]}, p_2 o^l$$

$$\xrightarrow{C} \boxed{q' o^{ll} p_2^l, p_2 o^l}^{[2]}$$

$$\xrightarrow{C} q'$$

Functional (Partial) Composition

A partial composition $\xrightarrow{C}$ is a *partial composition* ($\xrightarrow{@}$) if the width of the result is not greater than the maximum widths of the arguments

A partial composition that is not functional :

$$\Gamma, \boxed{q' o^{ll} q^l, q p_2^l \pi_2^l}^{[1]}, \Delta \xrightarrow{C} \Gamma, q' o^{ll} p_2^l \pi_2^l, \Delta$$

A functional composition :

$$\Gamma, \boxed{q' o^{ll} q^l, q o^l \pi_2^l}^{[2]}, \Delta \xrightarrow{@} \Gamma, q' \pi_2^l, \Delta$$

Parsing Using $\xrightarrow{@}$

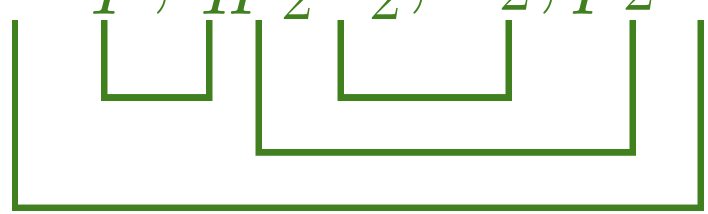
Theorem : $v_1 \cdots v_n \in \mathcal{L}(G)$ iff
for $1 \leq i \leq n$, $\exists X_i \in \mathcal{R}_{\underrightarrow{GCON}C^*}(I(v_i))$, and $\exists s' \in Pr$ such that:

$$\begin{cases} X_1, \cdots, X_n \xrightarrow{@^*} s' \\ s' \leq_{Pr} s \end{cases}$$

where $\mathcal{R}_{\underrightarrow{GCON}C^*}(E)$ is the completion of E using $\xrightarrow{GCON^*}$

Proof: property of (planar) pregroup nets : $\exists$ a type only connected to its immediate neighbours

$$q' o^{ll} q^l, q p_2^l \pi_2^l, \pi_2, p_2 o^l$$



Example

$$q'o^{ll}q^l, \boxed{qp_2^l\pi_2^l, \pi_2}^{[1]}, p_2o^l \xrightarrow{@} \boxed{q'o^{ll}q^l, qp_2^l}^{[1]}, p_2o^l$$

$$\xrightarrow{@} \boxed{q'o^{ll}p_2^l, p_2o^l}^{[2]}$$

$$\xrightarrow{@} q'$$

Parsing in n^3

Tabular Parsing Algorithm

For a grammar G and a list of words $v_1, \cdots, v_n \in \Sigma^+$, we compute for $1 \leq i \leq j \leq n$, $T^G_{v_1,\cdots,v_n}(i,j) \subset Tp$, the types associated to the sublist of words $v_i, \cdots, v_j$ using functional composition :

$$i = j : \quad T^G_{v_1,\cdots,v_n}(i,j) = \mathcal{R}_{G\underrightarrow{CON}C*}(I(v_i))$$

$$i < j : \quad T^G_{v_1,\cdots,v_n}(i,j) = \bigcup_{k=i}^{j-1} \left\{ Z \;\middle|\; \begin{array}{l} \exists X \in T^G_{v_1,\cdots,v_n}(i,k) \\ \exists Y \in T^G_{v_1,\cdots,v_n}(k+1,j) \\ \quad X, Y \xrightarrow{@} Z \end{array} \right\}$$

LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE

UNIVERSITÉ DE NANTES

Parsing in n^3

$v_1, \cdots, v_n \in \mathcal{L}(G)$ iff
$\exists s' \in Pr$ such that $s' \in T^G_{v_1,\cdots,v_n}(1,n)$ and $s' \leq_{Pr} s$

Algorithm :

1. Find the types associated to words
2. Add types using $\xrightarrow{GCONC^+}$
3. Compute recursively the types associated to a segment using $\xrightarrow{@}$
4. Search if s or $x \leq_{Pr} s$ is associated to the sentence.

Example (steps 1 and 2)

Parsing of “whom have you seen ?”

1. Lexicon:

$$\begin{array}{|lcl} \text{whom} & \mapsto & \{q' o^{ll} q^l\} \\ \text{have} & \mapsto & \{q p_2^l \pi_2^l\} \\ \text{you} & \mapsto & \{\pi_2\} \\ \text{seen} & \mapsto & \{p_2 o^l\} \end{array}$$

2. Types deduced using $\stackrel{GCONC^+}{\longrightarrow}$: nothing

LINA LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE
UNIVERSITÉ DE NANTES

Example (step 3)

Parsing of “whom have you seen ?”

3. Types associated to segments:

- Lenght = 1:

whom	have	you	seen
$\{q'o^{ll}q^l\}$	$\{qp_2^l\pi_2^l\}$	$\{\pi_2\}$	$\{p_2o^l\}$

- Length = 2:

whom have	have you	you seen
$\emptyset$	$\{qp_2^l\}$	$\emptyset$

- Length = 3:

whom have you	have you seen
$\{q'o^{ll}p_2^l\}$	$\{qo^l\}$

- Length = 4:

whom have you seen
$\{q'$ and $q'o^{ll}o^l\}$

LINA LABORATOIRE D'INFORMATIQUE DE NANTES ATLANTIQUE
UNIVERSITÉ DE NANTES

Example (step 4)

Parsing of “whom have you seen ?”

4. Atomic types : q' and $q' \leq_{Pr} s$
   $\Longrightarrow$ the sentence is correct