

## Introducción al Procesamiento de Lenguaje Natural

### Diciembre de 2010

Consideraciones generales
i) La prueba es sin material escrito.
ii) Escriba nombre y C.I. en todas las hojas.
iii) Numere todas las hojas.
iv) En la primera hoja, indique el total de hojas.
v) Comience cada ejercicio en una hoja nueva.
vi) Utilice las hojas de un solo lado.
vii) Entregue los ejercicios en orden
viii) El total de puntos es 80

#### Ejercicio 1 [ puntos]

1. Represente en lógica las siguientes oraciones:
  - a) Juan y Pedro corren.
  - b) Juan canta y baila.
  - $\text{corre}(j) \wedge \text{corre}(p)$
  - $\text{canta}(j) \wedge \text{baila}(j)$
  
2. Se propone la siguiente gramática (parcial) para dar cuenta de las expresiones:

REGLA	TIPO	SEMANTICA
$o \rightarrow gn\ gv$	t	$o.sem = gn.sem(gv.sem)$
$gn \rightarrow npropio$	$\langle\langle e,t \rangle, t \rangle$	$gn.sem = npropio.sem$
$gv \rightarrow v$	$\langle e,t \rangle$	$gv.sem = v.sem$
$cgn \rightarrow conj\ gn$	?	?
$gn \rightarrow gn\ cgn$	?	?
$cgv \rightarrow conj\ gv$	?	?
$gv \rightarrow gv\ cgv$	?	?
$npropio \rightarrow \text{Juan}$	$\langle\langle e,t \rangle, t \rangle$	$\lambda P.P(j)$
$npropio \rightarrow \text{Pedro}$	$\langle\langle e,t \rangle, t \rangle$	$\lambda P.P(p)$
$v \rightarrow \text{corren}$	$\langle e,t \rangle$	$\lambda x.corren(x)$
$v \rightarrow \text{canta}$	$\langle e,t \rangle$	$\lambda x.canta(x)$
$v \rightarrow \text{baila}$	$\langle e,t \rangle$	$\lambda x.baila(x)$

- i. Agregue regla(s) preléxica(s) para la conjunción “y”, indicando su tipo y valor semántico. Notar que todas las reglas son binarias, la conjunción se aplica primero al argumento que le sigue en el texto.

La conjunción “y” combina dos elementos de un mismo tipo X (en 2 pasos) y da un elemento de ese tipo. Dado que la utilizamos para tipos distintos de constituyentes debemos definir tipos semánticos distintos para la conjunción “y”

conj --> y	<<<e,t>,t>,<<e,t>,t>,<<e,t>,t>>>	$\lambda P.\lambda Q.\lambda x (P(x) \wedge Q(x))$
conj --> y	<<e,t>,<<e,t>,<e,t>>	$\lambda P.\lambda Q.\lambda x (P(x) \wedge Q(x))$

- ii. Complete la gramática propuesta en las partes donde aparece “?”.

cgn --> conj gn	<<<e,t>,t>,<<e,t>,t>>	cgn.sem=conj.sem(gn.sem)
gn --> gn cgn	<<e,t>,t>	gn.sem=cgn.sem(gn.sem)
cgv --> conj gv	<<e,t>,<e,t>>	cgv.sem=conj.sem(gv.sem)
gv --> gv cgv	<e,t>	gv.sem=cgv.sem(gv.sem)

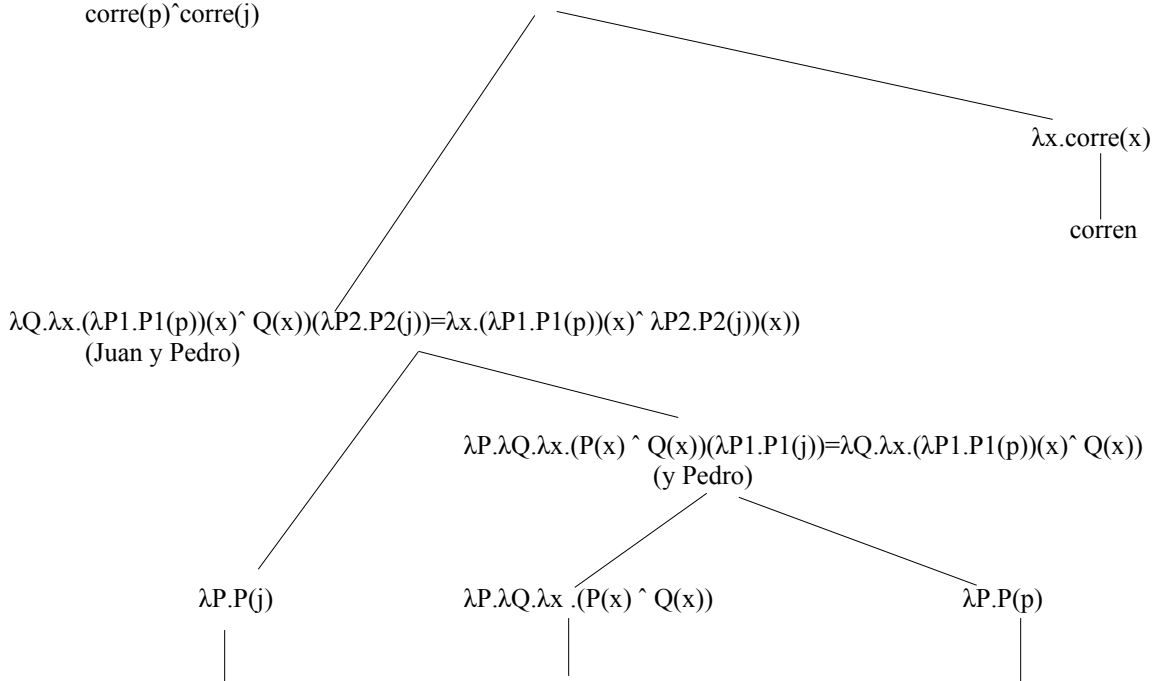
- iii. Realice la derivación completa para los ejemplos del punto 1).

1. Juan y Pedro corren.

$$\lambda x.(\lambda P1.P1(p))(x) \wedge \lambda P2.P2(j))(x)(\lambda x1.corre(x1)) =$$

$$\lambda P1.P1(p)(\lambda x1.corre(x1)) \wedge \lambda P2.P2(j)(\lambda x1.corre(x1)) = \lambda x1.corre(x1)(p) \wedge \lambda x1.corre(x1)(j) =$$

$$corre(p) \wedge corre(j)$$



Juan

y

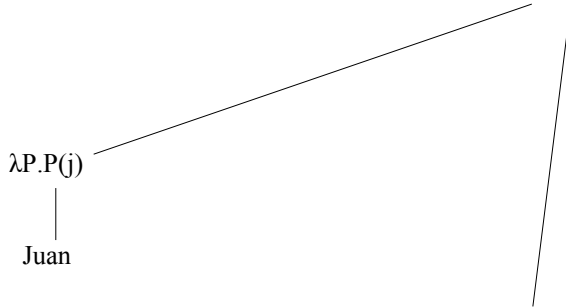
Pedro

2. Juan canta y baila

$$\lambda P.P(j)(\lambda x.(baila(x) \wedge canta(x))) =$$

$$\lambda x.(baila(x) \wedge canta(x))(j) = baila(j) \wedge canta(j)$$

(Juan canta y baila)

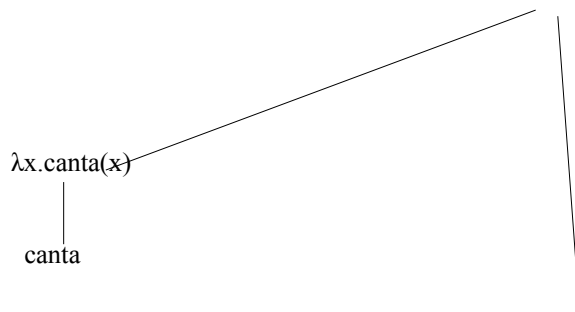


$$\lambda Q.\lambda x.(baila(x) \wedge Q(x))(\lambda x1.canta(x1)) =$$

$$\lambda x.(baila(x) \wedge \lambda x1.canta(x1)(x)) =$$

$$\lambda x.(baila(x) \wedge canta(x))$$

(canta y baila)

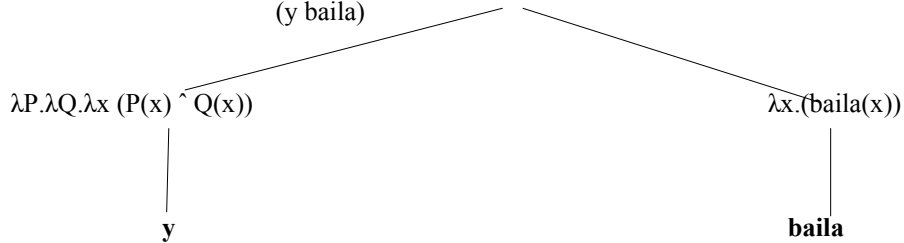


$$\lambda P.\lambda Q.\lambda x.(P(x) \wedge Q(x))(\lambda x1.(baila(x1))) =$$

$$\lambda Q.\lambda x.(\lambda x1.(baila(x1)))(x) \wedge Q(x) =$$

$$\lambda Q.\lambda x.(baila(x) \wedge Q(x))$$

(y baila)



Ejercicio 2 [ puntos]

1. Describa brevemente las ventajas y desventajas de aplicar estrategias puramente *top-down* o *bottom-up* para el análisis sintáctico.

Por un lado, las estrategias Top-Down solo generan árboles que comienzan en el símbolo inicial, lo cual es una ventaja sobre las estrategias Bottom-Up que exploran árboles que nunca llegarán al símbolo inicial.

Por otro lado, las estrategias Top-Down pierden tiempo explorando árboles que no son consistentes con la entrada, mientras que las Bottom-Up nunca exploran árboles que no sean consistentes, al menos localmente, con la entrada.

2. Considere la siguiente gramática (muy simplificada).

O	→	GV
GV	→	V GN   V GN GP
GN	→	N   N GP
GP	→	P GN
N	→	tomo   agua   manzana   manzanas   Pepe   María
V	→	tomo   toma   tomamos   toman
P	→	de   con   para

Aplique el algoritmo Early para la entrada *Tomo agua de manzana*.

chart[0]

$\gamma \rightarrow \cdot O$	[0,0]	
$O \rightarrow \cdot GV$	[0,0]	(predecir)
$GV \rightarrow \cdot V GN$	[0,0]	(predecir)
$GV \rightarrow \cdot V GN GP$	[0,0]	(predecir)

chart[1]

$V \rightarrow \text{tomo} \cdot$	[0,1]	(buscar)
$GV \rightarrow V \cdot GN$	[0,1]	(completar)
$GV \rightarrow V \cdot GN GP$	[0,1]	(completar)
$GN \rightarrow \cdot N$	[1,1]	(predecir)
$GN \rightarrow \cdot N GP$	[1,1]	(predecir)

chart[2]

$N \rightarrow \text{agua} \cdot$	[1,2]	(buscar)
$GN \rightarrow N \cdot$	[1,2]	(completar)
$GN \rightarrow N \cdot GP$	[1,2]	(completar)
$GV \rightarrow V GN \cdot$	[0,2]	(completar)
$GV \rightarrow V GN \cdot GP$	[0,2]	(completar)
$GP \rightarrow \cdot P GN$	[2,2]	(predecir)
$O \rightarrow GV \cdot$	[0,2]	(completar)
$\gamma \rightarrow O \cdot$	[0,2]	(completar) [no consumí toda la tira]

chart[3]

$P \rightarrow \text{de} \cdot$	[2,3]	(buscar)
$GP \rightarrow P \cdot GN$	[2,3]	(completar)
$GN \rightarrow \cdot N$	[3,3]	(predecir)
$GN \rightarrow \cdot N GP$	[3,3]	(predecir)

chart[4]

$N \rightarrow \text{manzana} \cdot$	[3,4]	(buscar)
$GN \rightarrow N \cdot$	[3,4]	(completar)
$GN \rightarrow N \cdot GP$	[3,4]	(completar)
$GP \rightarrow P GN \cdot$	[2,4]	(completar)
$GP \rightarrow \cdot P GN$	[4,4]	(predecir) [no quedan más elementos en la tira]
$GN \rightarrow N GP \cdot$	[1,4]	(completar)
$GV \rightarrow V GN GP \cdot$	[0,4]	(completar)

- GV → V GN • [0, 4] (completar)
- GV → V GN • GP [0, 4] (completar)
- O → GV • [0, 4] (completar)
- Y → O • [0, 4] (completar)

**Ejercicio 3 [ puntos]**

1. Describa brevemente el modelo del canal ruidoso y muestre una aplicación.

*El modelo del canal ruidoso supone que una observación corresponde a un string origen, que ha pasado por un canal con ruido que la deformó. Para textos, busca la palabra  $w$  que maximice  $P(O|w)$ . Para eso utiliza una aproximación Bayesiana, calculando  $P(w|O)P(w)$ .*

2. Se desea aplicar un modelo HMM para resolver la asignación de categorías léxicas a un texto. Del análisis del corpus de entrenamiento, se estiman las siguiente probabilidades:

Probabilidades de transición  $P(\text{tag}_j | \text{tag}_i)$

	<b>j:</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
<b>i</b>		</s>	PREP	DET	V	NN
0	<s>	0	0.2	0,05	0,1	0,1
1	PREP	0	0	0,5	0,1	0,1
2	DET	0	0	0	0	0,1
3	V	0,3	0,1	0,1	0,2	0,2
4	NN	0,2	0,2	0,1	0,1	0,3

Probabilidades de emisión:

- $P(\text{sobre} | \text{PREP}) = 0,07$
- $P(\text{sobre} | \text{V}) = 0,0012$
- $P(\text{sobre} | \text{N}) = 0,00062$
- $P(\text{el} | \text{DET}) = 0,4$

Aplice el algoritmo de Viterbi para obtener la secuencia de tags más probable para la expresión "sobre el sobre"

Aplicamos Viterbi:

$j$		0	1 (sobre)	2 (el)	3 (sobre)	4
0	<s>	1	0	0	0	$0,00000001736 * P(</s>   \text{NN}) = \dots$
1	PREP	0	$P(\text{sobre}   \text{PREP}) * P(\text{PREP}   <S>) = 0,0014/0$	0	0	0

2	DET	0	0	$0,0014 * P(e DET) * P(DET PREP) = 0,00028/1$ $0,00012 * P(e DET) * P(DET V) = 0,000048/3$ $0,000062 * P(e DET) * P(DET NN)$ $0,00000248/4$	=	0	0
3	V	0	$P(\text{sobre} V) * P(V <S>) = 0,0012 * 0,1 =$ <b>0,00012/0</b>	0		0	0
4	NN	0	$P(\text{sobre} NN) * P(NN <S>) = 0,00062 * 0,1 =$ <b>0,000062/0</b>	0		$0,00028 * P(\text{sobre} NN) * P(NN DET) =$ <b>0,000000017</b> <b>36/2</b>	0

La secuencia de tags más probable es : **PREP DET NN**

3. ¿Cuál es la diferencia entre el algoritmo Viterbi y el algoritmo Forward para HMMs?

Ver teórico.