

Introducción al Procesamiento de Lenguaje Natural Diciembre de 2011

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

Son medidas para evaluar la recuperación.

Precision: mide el porcentaje de documentos recuperados que resultan relevantes con el tema de la pregunta, calculándose como el cociente entre los recuperados relevantes sobre todos los recuperados

Recall: mide el porcentaje de documentos recuperados que resultan relevantes con el tema de la pregunta, calculándose como el cociente entre los recuperados relevantes sobre todos los documentos relevantes. No está claro cual es el valor del denominador, ya que si el usuario conoce de antemano el número de documentos relevantes de la colección, ¿xq no los recupera todos?.

Medida F , combina P y R

$$F = \frac{2PR}{P + R}$$

La medida E , es variante de medida F , permite poner mayor énfasis en precisión o en recuperación
El valor de β controla el balance entre P y R

$$E = \frac{(1 + \beta^2)PR}{\beta^2 P + R}$$

El factor β es un factor de ponderación, que permite darle peso más a la P o a la R

$\beta = 1$: Pesos iguales (En este caso $E=F$).

$\beta > 1$: Mayor peso a precisión

$\beta < 1$: Mayor peso a recuperación

Ejercicio 2

1.

$$\hat{e} = \arg \max_e P(f | e)P(e)$$

2. Los tres componentes son:

Modelo de lenguaje: Utilizado para resolver $P(e)$, que es la componente que mide la fluidez. Se suelen utilizar modelos de n-gramas para construir este modelo.

Modelo de traducción: Utilizado para resolver $P(f|e)$, que es la componente que mide la adecuación. En el modelo de traducción más simple, es una tabla que indica la probabilidad de traducir una palabra del lenguaje origen en una palabra del lenguaje destino.

Decodificador: Es un algoritmo de búsqueda que intenta devolver los candidatos e más probables de ser traducción de f. Reduce el espacio de búsqueda de todo el lenguaje E para que el problema se vuelva tratable computacionalmente.

Ejercicio 3

1.

Aplicamos Forward:

j		0	1 (la)	2 (nota)	3 (pálida)	4
0	<s>	1	0	0	0	$0,0000096 * P(</s> ADJ) = 0,00000096$
1	PRON	0	$I * P(la PRON) * P(PRON <S>) = 0,005$	0	0	0
2	NOM	0	$I * P(la NOM) * P(NOM <S>) = 0,00004$	$P(nota NOM) * P(NOM PRON) * 0,005 + P(nota NOM) * P(NOM NOM) * 0,00004 + P(nota NOM) * P(NOM DET) * 0,16 = 0,02 * 0,01 * 0,005 + 0,02 * 0,1 * 0,0004 + 0,02 * 0,5 * 0,16 = 0,00160108$	0	0
3	DET	0	$I * P(la DET) * P(DET <S>) = 0,16$	0	0	0
4	V	0		$P(nota V) * P(V PRON) * 0,005 + P(nota V) * P(V NOM) * 0,0004 + P(nota V) * P(V DET) * 0,16 = 0,2 * 0,06 * 0,005 + 0,2 * 0,4 * 0,0004 + 0,2 * 0 = 0,0000632$	0	0
5	ADJ	0	0	0	$0,00160108 * P(palida ADJ) * P(ADJ N) + 0,0000632 * P(palida ADJ) * P(ADJ V) = 0,00160108 * 0,03 * 0,2 = 0,0000096$	

La probabilidad de la secuencia es 0,00000096

2. Mencione una forma de estimar la probabilidad de ocurrencia de un n-grama, si no fue visto nunca en el corpus de entrenamiento.

Ver teórico.

Ejercicio 4

a. El pronombre relativo *que* (en el caso que estamos considerando) toma como argumento una oración a la que le falta el sujeto (un gv) y devuelve algo que es del tipo de los adjetivos (toma como argumento un nominal y devuelve un nominal).

que $\rightarrow \langle \langle e,t \rangle \langle \langle e,t \rangle, \langle e,t \rangle \rangle \rangle$

que.sem = $\lambda Q.\lambda P.\lambda x.(P(x) \wedge Q(x))$

b. Se debe agregar reglas para construir la frase relativa y reglas de léxico para cubrir las palabras de los ejemplos.

frel \rightarrow pronrel gv	frel.sem = pronrel.sem(gv.sem)
n' \rightarrow n frel	n'.sem = frel.sem(n.sem)
pronrel \rightarrow que	pronrel.sem = $\lambda Q.\lambda P.\lambda x.(P(x) \wedge Q(x))$
v \rightarrow comió	v.sem = $\lambda P.\lambda x..P(\lambda y.comió(x,y))$
v \rightarrow desapareció	v.sem = $\lambda x.desapareció(x)$
n \rightarrow bombones	n.sem = $\lambda x.bombón(x)$

c.
1. Un niño que salta quiere a María.

1	$o \rightarrow gn\ gv$	$\lambda Q. \exists x (\text{niño}(x) \wedge \text{salta}(x) \wedge Q(x))$ $(\lambda y. \text{quiere}(y,m)) =$ $\exists x (\text{niño}(x) \wedge \text{salta}(x) \wedge \lambda y. \text{quiere}(y,m))$ $(x) =$ $\exists x (\text{niño}(x) \wedge \text{salta}(x) \wedge \text{quiere}(x,m))$
1.1	$gn \rightarrow \text{det } n'$	$\lambda P. \lambda Q. \exists x (P(x) \wedge Q(x)) (\lambda y. (\text{niño}(y) \wedge \text{salta}(y))) =$ $\lambda Q. \exists x (\lambda y. (\text{niño}(y) \wedge \text{salta}(y)) (x) \wedge Q(x)) =$ $\lambda Q. \exists x (\text{niño}(x) \wedge \text{salta}(x) \wedge Q(x))$
1.1.1	$\text{det} \rightarrow \text{un}$	$\lambda P. \lambda Q. \exists x (P(x) \wedge Q(x))$
1.1.2	$n' \rightarrow n\ \text{frel}$	$\lambda P. \lambda x. (P(x) \wedge \text{salta}(x)) (\lambda y. \text{niño}(y)) =$ $\lambda x. (\lambda y. \text{niño}(y) (x) \wedge \text{salta}(x)) =$ $\lambda x. (\text{niño}(x) \wedge \text{salta}(x))$
1.1.2.1	$n \rightarrow \text{niño}$	$\lambda x. \text{niño}(x)$
1.1.2.2	$\text{frel} \rightarrow \text{pronrel } gv$	$\lambda Q. \lambda P. \lambda x. (P(x) \wedge Q(x)) (\lambda y. \text{salta}(y)) =$ $\lambda P. \lambda x. (P(x) \wedge \lambda y. \text{salta}(y) (x)) =$ $\lambda P. \lambda x. (P(x) \wedge \text{salta}(x))$
1.1.2.2.1	$\text{pronrel} \rightarrow \text{que}$	$\lambda Q. \lambda P. \lambda x. (P(x) \wedge Q(x))$
1.1.2.2.2	$gv \rightarrow v$	$\lambda x. \text{salta}(x)$
1.1.2.2.2.1	$v \rightarrow \text{salta}$	$\lambda x. \text{salta}(x)$
1.2	$gv \rightarrow v\ gn$	$\lambda P. \lambda y. P(\lambda x. \text{quiere}(y,x)) (\lambda Q. Q(m)) = \lambda y. \lambda Q. Q(m) (\lambda x. \text{quiere}(x,y)) =$ $\lambda y. (\lambda x. \text{quiere}(x,y)) (m) =$ $\lambda x. \text{quiere}(x,m)$
1.2.1	$v \rightarrow \text{quiere } a$	$\lambda P. \lambda y. P(\lambda x. \text{quiere}(y,x))$
1.2.2	$gn \rightarrow \text{npropio}$	$\lambda P. P(m)$
1.2.2.1	$\text{npropio} \rightarrow \text{María}$	$\lambda P. P(m)$

•Una mujer que comió todos los bombones desapareció.
 (No se pedía en la prueba)

1	$o \rightarrow gn\ gv$	$\lambda Q. \exists x ((mujer(x) \wedge \forall z(bombón(z) \rightarrow comió(x,z)) \wedge Q(x)) (\lambda y. desapareció(y)) = \exists x ((mujer(x) \wedge \forall z(bombón(z) \rightarrow comió(x,z)) \wedge desapareció(x))$
1.1	$gn \rightarrow det\ n'$	$\lambda P. \lambda Q. \exists x (P(x) \wedge Q(x)) (\lambda w. (mujer(w) \wedge \forall z(bombón(z) \rightarrow comió(w,z))) = \lambda Q. \exists x (\lambda w. (mujer(w) \wedge \forall z(bombón(z) \rightarrow comió(w,z))) (x) \wedge Q(x)) = \lambda Q. \exists x ((mujer(x) \wedge \forall z(bombón(z) \rightarrow comió(x,z)) \wedge Q(x))$
1.1.1	$det \rightarrow una$	$\lambda P. \lambda Q. \exists x (P(x) \wedge Q(x))$
1.1.2	$n' \rightarrow n\ frel$	$\lambda P. \lambda x. (P(x) \wedge \forall z(bombón(z) \rightarrow comió(x,z)) (\lambda y. mujer(y)) = \lambda x. (\lambda y. mujer(y) (x) \wedge \forall z(bombón(z) \rightarrow comió(x,z))) = \lambda x. (mujer(x) \wedge \forall z(bombón(z) \rightarrow comió(x,z)))$
1.1.2.1	$n \rightarrow mujer$	$\lambda x. mujer(x)$
1.1.2.2	$frel \rightarrow pronrel\ gv$	$\lambda Q. \lambda P. \lambda x. (P(x) \wedge Q(x)) (\lambda w. \forall z(bombón(z) \rightarrow comió(w,z)) = \lambda P. \lambda x (P(x) \wedge \lambda w. \forall z(bombón(z) \rightarrow comió(w,z)) (x)) = \lambda P. \lambda x (P(x) \wedge \forall z(bombón(z) \rightarrow comió(x,z)))$
1.1.2.2.1	$pronrel \rightarrow que$	$\lambda Q. \lambda P. \lambda x. (P(x) \wedge Q(x))$
1.1.2.2.2	$gv \rightarrow v\ gn$	$\lambda P. \lambda x. P(\lambda y. comió(x,y)) (\lambda Q. \forall z(bombón(z) \rightarrow Q(z))) = \lambda x. \lambda Q. \forall z(bombón(z) \rightarrow Q(z)) (\lambda y. comió(x,y)) = \lambda x. \forall z(bombón(z) \rightarrow \lambda y. comió(x,y)) (z) = \lambda x. \forall z(bombón(z) \rightarrow comió(x,z))$
1.1.2.2.2.1	$v \rightarrow comió$	$\lambda P. \lambda x. P(\lambda y. comió(x,y))$
1.1.2.2.2.2	$gn \rightarrow det\ n'$	$\lambda P. \lambda Q. \forall x (P(x) \rightarrow Q(x)) (\lambda y. bombón(y)) = \lambda Q. \forall x (\lambda y. bombón(y)(x) \rightarrow Q(x)) = \lambda Q. \forall x (bombón(x) \rightarrow Q(x))$
1.1.2.2.2.2.1	$det \rightarrow todos_los$	$\lambda P. \lambda Q. \forall x (P(x) \rightarrow Q(x))$
1.1.2.2.2.2.2	$n' \rightarrow n$	$\lambda x. bombón(x)$
1.1.2.2.2.2.2.1	$n \rightarrow bombones$	$\lambda x. bombón(x)$
1.2	$gv \rightarrow v$	$\lambda x. desapareció(x)$
1.2.1	$v \rightarrow desapareció$	$\lambda x. desapareció(x)$

d. En el ejemplo

3. *El libro que leyó Juan desapareció.*

la frase relativa es *que leyó Juan* y la diferencia esencial es que el pronombre relativo *que* ocupa la posición complemento de objeto en la frase relativa.

Para incorporar este caso a la gramática desarrollada hasta el momento se debe:

- a) considerar una estructura de oración con el sujeto después del verbo (el libro que leyó Juan), y definir la reglas de sintaxis / semántica necesarias
- b) definir un tipo semántico alternativo para el pronombre relativo *que*, que se agregaría al ya existente