

# 1. Recapitulación

Para la clase de hoy, conviene tener presente los siguientes conceptos, ya abordados en las clases anteriores:

- **Conjuntos:** colección no ordenada de elementos. En particular, nos interesan las siguientes nociones:
  - **Inclusión:**
    - **Impropia:**  $A \subseteq B$ , los miembros de  $A$  también son miembros de  $B$ , que a su vez puede o no tener otros miembros.
    - **Propia:**  $A \subset B$ , los miembros de  $A$  también son miembros de  $B$ , que además tiene otros miembros.
  - **Unión:**  $A \cup B$ , la unión entre dos conjuntos devuelve un nuevo conjunto que contiene los elementos de ambos.
- **Funciones:** relación particular que se da entre dos conjuntos; es una operación que toma elementos de un conjunto y los empareja con un único elemento del otro conjunto.
  - **Tipos de funciones:**
    - **Total:** todos los elementos del conjunto de salida (*input*) son alcanzados por la relación.
    - **Parcial:** no todos los elementos del conjunto de salida son alcanzados por la relación.
  - **Propiedades:**
    - **Inyectividad:** cada elemento del conjunto de llegada se corresponde con un único elemento del conjunto de salida.
    - **Sobreyectividad:** todos los elementos del conjunto de salida son alcanzados por la relación (*i.e.*  $\mathbb{I} = \mathbb{C}$ ).
- **Noción de rasgo** La noción de rasgo, uno de los ejes centrales de esta clase, ya apareció en la clase 5 al hablar de GPSG. Las gramáticas de estructura de frases generalizadas reemplazan los nodos no terminales por conjuntos formados por rasgos. Es decir, conjuntos de pares  $\langle \text{atributo, valor} \rangle$ .

Esta clase está basada en la siguiente bibliografía:

- Bird, S., Klein, E., y Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. “O’Reilly Media, Inc.”. “Chapter 9: Building Feature-Based Grammars”, pp. 327–360.
- Blevins, J. P. (2011). Feature-based grammar. En *Non-Transformational Syntax*, pp. 297–324. Wiley Blackwell, Massachusetts.
- Moreno Sandoval, A. (2001). *Gramáticas de Unificación y Rasgos*. Antonio Machado, Madrid. “Capítulo 3: El uso de rasgos en la descripción gramatical” y “Capítulo 4: La operación de unificación”, pp. 41–78.
- Solias Arís, M. T. (2015). *Métodos formales en Lingüística*. Síntesis. “Capítulo 3: Aplicaciones de las Gramáticas Formales y de las matrices de rasgos”, pp. 95–144.

## 2. Gramáticas basadas en rasgos

Las gramáticas basadas en rasgos engloban un conjunto de gramáticas que comparten entre sí la noción de rasgo como elemento lingüístico mínimo. Su intención es la de aumentar el poder descriptivo de las gramáticas, pero evitando limitaciones presentes en las gramáticas transformacionales tales como:

- ser indecibles<sup>1</sup>;
- no poder dar cuenta de fenómenos lingüísticos tales como el mando-c o las dependencias cruzadas<sup>2</sup>;

---

<sup>1</sup>“The mathematical properties of the baroque systems are almost entirely unknown: we are ignorant, for example, as to whether ungrammaticality with respect to such grammars is decidable”(Gazdar 1982:133)

<sup>2</sup>“Una de las configuraciones arbóreas que ha recibido un uso más dilatado en la bibliografía de la investigación en lingüística formal es el denominado mando-c, o mando categorial; [...] Por supuesto, este tipo de operaciones exceden el poder expresivo de las Gramáticas Formales de la Jerarquía de Chomsky, pero hay que tener en cuenta que Chomsky nunca consideró su teoría de gramáticas formales como una teoría apropiada para la representación de los fenómenos lingüísticos de las lenguas naturales” (Solias Arís 2015:100)

- ser no recursivas, por lo que la adquisición no sería posible.<sup>3</sup>

Algunos elementos que comparten estas gramáticas son su postura lexicalista; la descripción a las dependencias locales y no locales como un problema de compatibilidad de la información compartida por los rasgos descritos en la gramática; y una perspectiva en la que la compatibilidad en la información de los rasgos no es solo un chequeo de compatibilidad de los dependientes, sino el surgimiento de un objeto que de alguna manera combina los rasgos de sus dependientes.

### 3. Aproximación lexicalista

Las gramáticas basadas en rasgos nos proponen un abordaje lexicalista para la descripción lingüística. Aquí, es el léxico el que permite hacer generalizaciones y posibilita la existencia de estructuras gramaticales. Las reglas sintácticas, en cambio, tienen un papel menor y son consideradas mucho más simples en comparación con las especificaciones léxicas.

De todos modos, esta definición de lexicalismo no implica que o que se combina por medio de la sintaxis sean palabras ya consituadas necesariamente. Esto podría ser así (enfoque sostenido por GPSG o HPSG) o no (enfoque minimalista).

Ni tampoco debe confundirse con la idea de que solo tenemos especificaciones léxicas y carecemos en absoluto de reglas sintácticas (propuesta adoptada por las gramáticas categoriales).

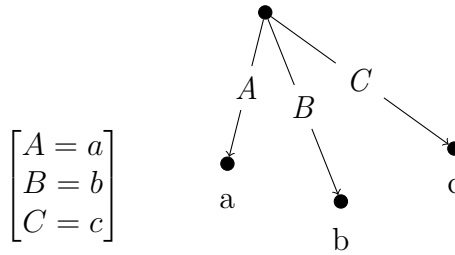
### 4. Estructuras de rasgos

Las gramáticas basadas en rasgos tratan las propiedades morfo-sintácticas como haces de rasgos que pueden ser representados por medio de matrices atributo-valor (AVM) o por medio de grafos dirigidos (ver figura 1).

Estas estructuras pueden representar diferentes clases y tener distintos usos en cada gramática particular, pero todas comparten los elementos descriptivos. Dichas estructuras no están limitadas

---

<sup>3</sup>(Gazdar 1982:133)



(a) En una matriz      (b) En un grafo

Figura 1: Representación de estructuras de rasgo

en la cantidad de pares que pueden contener y su tamaño va a depender del nivel de detalle con el que la gramática provea a la propiedad lingüística definida.

Si estas estructuras consisten únicamente en rasgos atómicos, que no pueden subdividirse, serán estructuras atómicas. Si, en cambio, contienen algún atributo con valores complejos (*i.e.* los valores también son pares atributo–valor), serán estructuras complejas.

Ver ejercicios Otra forma de pensar estas estructuras es como funciones parciales que toman un atributo y devuelven su valor asociado. Si, por ejemplo, llamamos  $S$  a la estructura definida en 1a, podemos pensar que  $S(A)$  como la función que toma el atributo  $A$  y devuelve su valor en la estructura  $S$ .

## 4.1. El rasgo

Como ya vimos, el rasgo (*feature*) es un par  $\langle \text{atributo, valor} \rangle$  que describe una propiedad de la gramática. Especifiquemos algunos rasgos con su atributo y sus posibles valores:

- GEN : fem, masc
- PER : 1, 2, 3
- PLU : +, -

Al hacer uso de cualquiera de ellos, el valor deberá seleccionarse, ya que no puede ser más de uno al mismo tiempo. Es decir, el rasgo TIEMPO no podrá adquirir el valor “PRESENTE” y “PASADO” simultáneamente en una oración. También aquí podemos notar que,

para cada rasgo, el conjunto de valores puede pertenecer a diferentes tipos de datos. En estos ejemplos podemos decir que valores como “fem” o “masc” son de tipo cadena (*string*), 1, 2, y 3, de tipo *integer*; y finalmente + o -, de tipo booleano. Este tipo de dato no va a ser igual para los mismos rasgos en todas las gramáticas. El número es indicado en algunas como una cadena (NUM: sg, pl).

A su vez, los valores que puede tomar un rasgo no necesariamente pertenecen a un tipo de dato simple, sino que pueden tomar la forma de una estructura compuesta por otros rasgos. Por ejemplo, el rasgo COOR (coordinación) puede tomar como valor un conjunto de rasgos  $\left( \begin{smallmatrix} PER & 1 \\ PLU & + \end{smallmatrix} \right)$ .

¿Cuál es el atributo en el par que compone al rasgo COOR?

Las gramáticas basadas en rasgos van a utilizar a este objeto como el elemento mínimo de la gramática y por medio de rasgos van a representar estructuras sintagmáticas, restricciones gramaticales y entradas léxicas.

#### 4.1.1. Estructuras sintagmáticas

Especifiquemos los rasgos que necesitamos para dar cuenta de la concordancia en español.

$$\begin{aligned}
 SN \left[ \begin{smallmatrix} NUM = \alpha \\ GEN = \beta \\ PER = \gamma \end{smallmatrix} \right] &\rightarrow DET \left[ \begin{smallmatrix} NUM = \alpha \\ GEN = \beta \\ PER = \gamma \end{smallmatrix} \right] N \left[ \begin{smallmatrix} NUM = \alpha \\ GEN = \beta \\ PER = \gamma \end{smallmatrix} \right] \\
 O \left[ \begin{smallmatrix} NUM = \alpha \\ PER = \gamma \end{smallmatrix} \right] &\rightarrow SN \left[ \begin{smallmatrix} NUM = \alpha \\ PER = \gamma \end{smallmatrix} \right] SV \left[ \begin{smallmatrix} NUM = \alpha \\ PER = \gamma \end{smallmatrix} \right]
 \end{aligned}$$

Con estas reglas ya podemos producir las oraciones “El león corre.”, “Las leonas corren.” sin necesidad de multiplicar reglas porque las categorías están descriptas ahora como estructuras de rasgos cuyos valores son variables. Al instanciar estas categorías en una oración real, “llenaremos” el valor con cualquiera de los permitidos para dicho atributo. Así será sencillo definir las oraciones gramaticales y las agramaticales como aquellos casos en los que la variable implicada en todas las estructuras toma el mismo valor o no.

¿Cuáles son los valores posibles para cada una de las variables presentes en la coordinación?

#### 4.1.2. Restricciones gramaticales

Las dependencias no locales también van a poder definirse en términos de rasgos.

Volvamos al ejemplo que rompía la *non-tangling condition*:

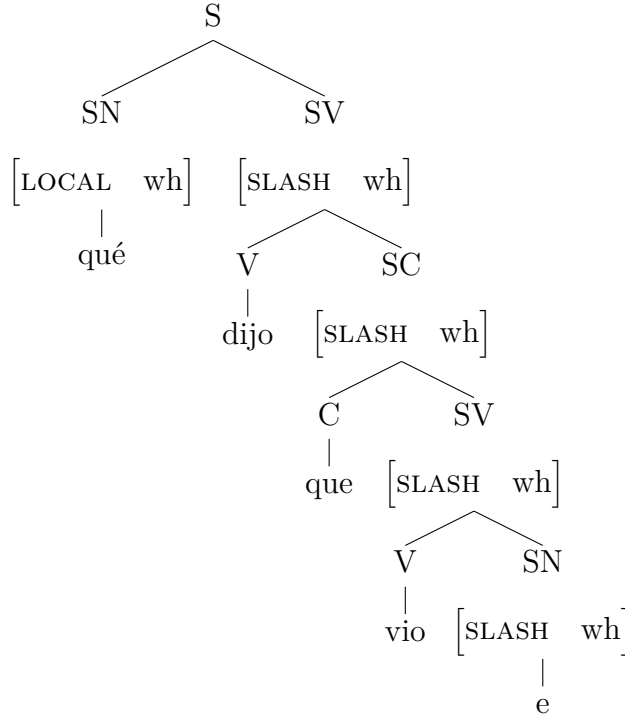
- ¿Qué dijo que vio?

En esta oración, el objeto del verbo “vio” es el interrogativo “qué” que aparece al comienzo de la oración. Este tipo de dependencia no local va a ser tratado por estas gramáticas por medio de *gaps* y *fillers*, ya sea desarmando las dependencia cruzada en una secuencia de dependencias locales por medio del uso del atributo SLASH (que veremos un poco más adelante) o de la indexación en estructuras-f en LFG.

En el ejemplo, el subíndice 1 indica que el último elemento de la cadena es un *gap* (hueco) y el primer elemento indexado, su *filler* (relleno).

- ¿Qué<sub>1</sub> dijo que vio <sub>1</sub>?

(1)



#### 4.1.3. Feature sharing

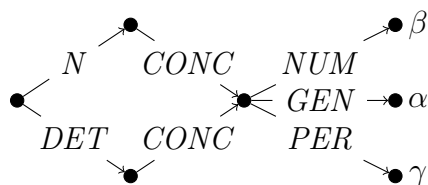
Según las reglas de reescritura vistas arriba, la concordancia en español es una estructura compuesta por más de un rasgo y se caracteriza por ser compartida por más de una categoría. Si pensamos en la estructura de rasgos de una categoría como el SN de nuestras reglas, sabremos que estará compuesto por un N y un DET que necesariamente comparten el atributo CONC y, recursivamente, deben coincidir en todos los valores que toman a su vez los rasgos que componen a CONC. Esto indica que las dos estructuras de rasgos (N y DET) comparten (*share*) un rasgo común (CONC).

$$\left[ \begin{array}{c} CAT \\ N \\ DET \end{array} \left[ \begin{array}{c} SN \\ CONC \boxed{1} \\ [CONC \boxed{1}] \end{array} \left[ \begin{array}{c} NUM = \alpha \\ GEN = \beta \\ PER = \gamma \end{array} \right] \right] \right]$$

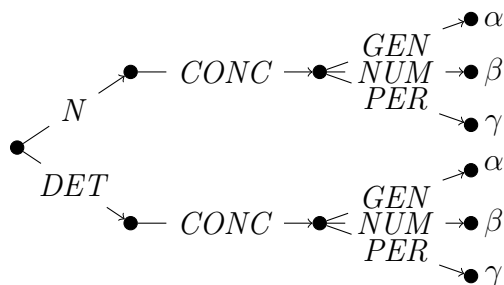
El que compartan este rasgo no es obra de la casualidad, sino que

es un elemento estructural del SN. Observar si esta característica, también denominada reentrancia, está o no presente en la estructura resulta más visible al representar las estructuras de rasgos como grafos.

### Rasgos compartidos



### Rasgos divididos



#### 4.1.4. El léxico

En estas gramáticas, las mismas estructuras de rasgos serán las que defininan los items del léxico.

$$\left[ \begin{array}{l} SIGNIFICANTE = torta \\ LEXEMA = torta \\ CAT = N \\ GEN = fem \\ PLU = - \end{array} \right]$$

¿Cuál sería la estructura de rasgos del item léxico “leona”?

Ver  
ejercicios  
en 9.3.



## 5. Métodos para abreviar la codificación

Dado que estas gramáticas implican una alta complejidad en la definición de las piezas léxicas, resulta de utilidad contar con ciertos métodos que faciliten su confección y mantenimiento. Estos, a su vez, permitirán evitar, en los casos que sea posible, la redundancia en la especificación de la información.

GPSG, por ejemplo, recurre a las siguientes técnicas:

1. **Convenciones de abreviación:** utilizar códigos para distinguir estructuras específicas. Por ejemplo: “dar” se especifica  $V[3]$ , con lo cual se reemplaza una expresión más compleja como  $[[N-][V+][SUBCAT[NP, PP[a]]]]$
2. **Restricciones de coaparición de rasgos:** especificar rasgos incompatibles para determinadas entradas léxicas con el fin de evitar que se generen categorías mal formadas. Por ejemplo, un N que denota una entidad animada no puede recibir el rasgo inanimado.
3. **Especificación de rasgos por defecto:** definición de rasgos por defecto de modo que solo sea necesario escribir los rasgos marcados, lo que posibilita tener que escribirlo una menor cantidad de veces.

## 6. La noción de herencia

Un punto que puede resultarnos de interés es que el concepto de matrices de rasgos y unificación fue inicialmente introducido dentro de la lingüística computacional. De aquí que algunos conceptos resulten familiares a la programación orientada a objetos. Lo vimos en los tipos de datos que describen los valores de los rasgos y podemos nuevamente verlo en el concepto de herencia.

En el caso de estas gramáticas, en las que las entradas léxicas se definen como haces de rasgos, la idea de herencia se desprende de la jerarquía que organiza el conocimiento léxico. Los niveles inferiores (o más concretos) heredarían por defecto (*default*) la información

contenida en los niveles superiores (o más abstractos) con la posibilidad de sobrecribir alguno de esos rasgos heredados.

El ejemplo que nos da Moreno Sandoval es bastante claro. Si en la jerarquía tenemos una entrada AVE con los rasgos  $\left(\begin{smallmatrix} plumas+ \\ vuela+ \end{smallmatrix}\right)$ , entradas como GAVIOTA heredarían automáticamente estos rasgos sin necesidad de repetirlos  $\left(\begin{smallmatrix} AVE \end{smallmatrix}\right)$  (herencia completa). Pero una entrada como PINGÜINO sobrecribiría uno de los rasgos:  $\left(\begin{smallmatrix} plumas+ \\ vuela- \end{smallmatrix}\right)$  (herencia normal). Alunas gramáticas adoptan uno o el otro tipo de herencia. Por ejemplo, HPSG únicamente asume el modo completo de herencia.

## 7. Unificación

La unificación ( $\sqcup$ ) es un proceso que nos permite combinar dos estructuras de rasgos en una única estructura que contenga la información especificada en ambas. El siguiente ejemplo, tomado de Blevins (2011), nos muestra el proceso de unificación entre el pronombre masculino *er* (‘él’) y el requerimiento del sujeto del verbo *singt* (‘canta’):

$$\left[ \begin{array}{c} \text{per 3} \\ \text{num SG} \\ \text{gen MASC} \\ \text{caso NOM} \end{array} \right] \sqcup \left[ \begin{array}{c} \text{per 3} \\ \text{num SG} \\ \text{caso NOM} \end{array} \right] = \left[ \begin{array}{c} \text{per 3} \\ \text{num SG} \\ \text{gen MASC} \\ \text{caso NOM} \end{array} \right]$$

Este procedimiento es muy similar a la operación de unión entre conjuntos. Sin embargo, lo separa de ésta una diferencia crucial: mientras que la unión combina dos conjuntos y nos devuelve un nuevo conjunto, la unificación combina dos estructuras y nos devuelve una nueva estructura donde los pares atributo–valor deben estar asociados por una función (*i.e.* cada atributo debe estar asociado a un único valor).

Analicemos un ejemplo tomado de Moreno Sandoval (2001). Supongamos que tenemos las estructuras definidas en 2, correspondientes al sintagma nominal “virus mutantes”:

(2)

$$N = \left[ \begin{array}{l} \text{categoría} = N \\ \text{concordancia} = \left[ \begin{array}{l} \text{núm [ ]} \\ \text{gén masc} \end{array} \right] \end{array} \right]$$

$$ADJ = \left[ \begin{array}{l} \text{categoría} = ADJ \\ \text{concordancia} = \left[ \begin{array}{l} \text{núm plu} \\ \text{gén masc} \end{array} \right] \end{array} \right]$$

Si nuestra gramática tiene la regla sintagmática  $SN \rightarrow N ADJ$ , podremos unificar ambas estructuras tal como se ve en 3.

(3)

$$N \sqcup ADJ = \left[ \begin{array}{l} \text{categoría} = SN \\ \text{concordancia} = \left[ \begin{array}{l} \text{núm plu} \\ \text{gén masc} \end{array} \right] \end{array} \right]$$

Y si convertimos esta estructura en un conjunto, se verá del siguiente modo (4):

(4)

$$SN = \{ \langle \text{categoría}, SN \rangle, \\ \langle \text{concordancia}, \{ \langle \text{núm}, \text{plu} \rangle, \langle \text{gén masc} \rangle \} \} \}$$

Sin embargo, no llegaremos al mismo resultado si primero convertimos las estructuras de 2 en conjuntos (5) y luego aplicamos sobre estos una unión (3).

(5)

$$N = \{ \langle \text{categoría}, N \rangle, \\ \langle \text{concordancia}, \{ \langle \text{núm}, [ ] \rangle, \langle \text{gén}, \text{masc} \rangle \} \} \}$$

$$ADJ = \{ \langle \text{categoría}, ADJ \rangle, \\ \langle \text{concordancia}, \{ \langle \text{núm}, \text{plu} \rangle, \langle \text{gén}, \text{masc} \rangle \} \} \}$$

(6)

$$N \cup ADJ = \{ \langle \text{categoría, SN} \rangle, \\ \langle \text{concordancia, } \{ \langle \text{núm, [ ]} \rangle, \langle \text{gén masc} \rangle \} \rangle, \\ \langle \text{concordancia, } \{ \langle \text{núm, plu} \rangle, \langle \text{gén masc} \rangle \} \} \}$$

El resultado de este procedimiento (6) ya no puede ser reconvertido en una estructura de rasgos, dado que la concordancia ya no se encuentra mapeada de manera unívoca a un valor, sino que ahora mantiene una relación con dos valores<sup>4</sup>.

Los únicos casos en los cuales la unión genera el mismo conjunto que la unificación son aquellos en los cuales las estructuras de rasgos involucradas son de tipo atómico.

Para que la unificación entre dos matrices de rasgos pueda tener lugar, es necesario que estas sean compatibles entre sí. Si esto no sucede, la unificación fallará.

Ver La tabla 1, adaptada de Moreno Sandoval (2001), muestra los posibles ejercicios situaciones de compatibilidad e incompatibilidad que pueden darse entre dos estructuras  $E1$  y  $E2$ .

Tabla 1: Estructuras compatibles e incompatibles

Compatibilidad	Incompatibilidad
$E1$ y $E2$ no tienen ningún atributo en común.	
En alguna de las dos estructuras, los atributos presentes en la otra se encuentran subespecificados.	$E1$ le asigna un valor atómico a un atributo y $E2$ , uno complejo.
$E1$ y $E2$ asignan el mismo valor atómico a determinado atributo.	$E1$ y $E2$ asignan valores atómicos distintos a un mismo atributo.
$E1$ y $E2$ asignan el mismo valor complejo a determinado atributo.	$E1$ y $E2$ asignan valores complejos incompatibles a un mismo atributo.

<sup>4</sup>Recordar la definición de funciones vista en la página 1.

## 8. Subsunción

La subsunción ( $\sqsubseteq$ ) es una relación que puede establecerse entre dos estructuras cuando una se encuentra menos especificada que la otra.

Como hemos visto anteriormente, no todas las estructuras cuentan con la misma cantidad de información. Es posible encontrarnos con estructuras que no están especificadas para determinado atributo y obtienen esa especificación al unificarse con otra. En estos casos, la estructura subespecificada (y más general) subsume a la estructura resultante (más específica).

Esta relación puede asociarse con la noción de herencia vista previamente. Si una estructura  $B$  hereda los rasgos de otra estructura  $A$ , sin sobrescribir ninguno de ellos, diremos que  $A$  *subsume* a  $B$ .

Moreno Sandoval (2001) nos propone un buen ejercicio: pensar estructuras de rasgos con distintos niveles de especificación e indicar la relación de subsunción entre ellas. Ver ejercicios en 9.5.

### ■ Estructuras de rasgos

- $E_{var} = []$
- $E_{SNsg3} = \begin{bmatrix} \text{cat SN} \\ \text{concordancia} \begin{bmatrix} \text{núm SING} \\ \text{pers 3p} \end{bmatrix} \end{bmatrix}$
- $E_{SNsg3Suj} = \begin{bmatrix} \text{cat SN} \\ \text{concordancia} \begin{bmatrix} \text{núm SING} \\ \text{pers 3p} \end{bmatrix} \\ \text{sujeto} \begin{bmatrix} \text{núm SING} \\ \text{pers 3p} \end{bmatrix} \end{bmatrix}$
- $E'_{SNsg3Suj} = \begin{bmatrix} \text{cat SN} \\ \text{concordancia} \begin{bmatrix} 1 \\ \text{núm SING} \\ \text{pers 3p} \end{bmatrix} \\ \text{sujeto} \begin{bmatrix} 1 \end{bmatrix} \end{bmatrix}$

### ■ Relación de subsunción

$$E_{var} \sqsubseteq E_{SNsg3} \sqsubseteq E_{SNsg3Suj} \sqsubseteq E'_{SNsg3Suj}$$

## 9. Ejercicios

### 9.1. Estructuras como funciones parciales

Viene de      Dadas las siguientes estructuras, indicá qué valor devuelven las  
página 4. funciones a continuación:

$$S = \begin{bmatrix} A & a \\ B & b \\ C & \begin{bmatrix} X & x \\ Y & y \end{bmatrix} \end{bmatrix} \quad P = \begin{bmatrix} A & a \\ B & \begin{bmatrix} X & x \\ Y & y \end{bmatrix} \end{bmatrix}$$

$$S(B)=$$

Tipo de valor:

$$P(A)=$$

Tipo de valor:

$$P(B)=$$

Tipo de valor:

$$S(C)=$$

Tipo de valor:

### 9.2. Estructuras sintagmáticas

Viene de      Escribí las reglas derivativas de una gramática que genere las  
página 5 siguientes oraciones (y no otras) utilizando rasgos:

- El león corre.
- La chita corre.
- Los leones corren.
- Las chitas corren.

### 9.3. Matrices de rasgos en el léxico

Construí las estructuras de rasgos para las siguientes entradas léxicas: Viene de página 8

- dio
- en
- yo

### 9.4. Unificación

#### 9.4.1. Posibilidad de unificación

Indicá si las siguientes estructuras pueden ser unificadas de manera exitosa o no. Si fallan, explicá por qué. Viene de página 12

1.

$$\begin{bmatrix} A = a \\ B = b \end{bmatrix} \sqcup \begin{bmatrix} A = a \\ C = c \end{bmatrix}$$

Unificación:

Justificación:

2.

$$\begin{bmatrix} A = a \\ B = b \end{bmatrix} \sqcup \begin{bmatrix} A = b \\ C = c \end{bmatrix}$$

Unificación:

Justificación:

3.

$$\begin{bmatrix} A = a \\ B = \begin{bmatrix} X = x \\ Y = y \end{bmatrix} \end{bmatrix} \sqcup \begin{bmatrix} A = a \\ B = \begin{bmatrix} X = z \\ Y = y \end{bmatrix} \end{bmatrix}$$

Unificación:

Justificación:

4.

$$\left[ \begin{array}{c} M = m \\ N = \left[ \begin{array}{c} X = x \\ Y = y \end{array} \right] \end{array} \right] \sqcup \left[ \begin{array}{c} M = m \\ N = n \end{array} \right]$$

Unificación:

Justificación:

5.

$$\left[ \begin{array}{c} M = m \\ N = \left[ \begin{array}{c} X = x \\ Y = ? \end{array} \right] \end{array} \right] \sqcup \left[ \begin{array}{c} M = m \\ N = \left[ \begin{array}{c} X = x \\ Y = y \end{array} \right] \end{array} \right]$$

Unificación:

Justificación:

6.

$$\left[ \begin{array}{c} A = a \\ B = \left[ \begin{array}{c} X = x \\ Y = y \end{array} \right] \end{array} \right] \sqcup \left[ \begin{array}{c} C = c \\ D = d \end{array} \right]$$

Unificación:

Justificación:

#### 9.4.2. Estructura resultante

A partir de las siguientes reglas y estructuras de rasgos, realizá las unificaciones indicadas.

■ Reglas sintácticas

- $SN \rightarrow (DET) N ADJ$
- $SV \rightarrow N$
- $SP \rightarrow P SN$

■ Estructuras de rasgos



- $A = \begin{bmatrix} \text{cat N} \\ \text{conc} \begin{bmatrix} \text{núm PL} \\ \text{gén FEM} \end{bmatrix} \end{bmatrix}$
- $B = \begin{bmatrix} \text{cat N} \\ \text{conc} \begin{bmatrix} \text{núm PL} \\ \text{per 3p} \end{bmatrix} \end{bmatrix}$
- $C = \begin{bmatrix} \text{cat V} \\ \text{tiem PAS} \end{bmatrix}$
- $D = [\text{cat ADJ}]$
- $E = \begin{bmatrix} \text{cat P} \\ \text{valor 'sin'} \end{bmatrix}$

1.  $A \sqcup B =$

2.  $A \sqcup C =$

3.  $A \sqcup D =$

4.  $E \sqcup (A \sqcup D) =$

## 9.5. Subsunción

¿Cuáles de las siguientes estructuras subsumen a otras?

(a)  $\begin{bmatrix} \text{nombre HERMINIA} \\ \text{apellido } \boxed{1} \text{ GARCÍA} \\ \text{padre} \begin{bmatrix} \text{nombre MARIO} \\ \text{apellido } \boxed{1} \end{bmatrix} \\ \text{madre} \begin{bmatrix} \text{nombre MIRIAM} \\ \text{apellido GONZÁLEZ} \end{bmatrix} \end{bmatrix}$

Viene de  
página  
**13**

$$(b) \left[ \begin{array}{l} \text{nombre CARLOS} \\ \text{apellido } \boxed{1} \text{ GARCÍA} \\ \text{padre } \left[ \begin{array}{l} \text{nombre MARIO} \\ \text{apellido } \boxed{1} \end{array} \right] \\ \text{madre } \left[ \begin{array}{l} \text{nombre MIRIAM} \\ \text{apellido GONZÁLEZ} \end{array} \right] \end{array} \right]$$

$$(c) \left[ \text{apellido GARCÍA} \right]$$

$$(d) \left[ \begin{array}{l} \text{nombre MARIO} \\ \text{apellido } \boxed{1} \text{ GARCÍA} \\ \text{padre } \left[ \begin{array}{l} \text{nombre EDUARDO} \\ \text{apellido } \boxed{1} \end{array} \right] \\ \text{madre } \left[ \begin{array}{l} \text{nombre CARMEN} \\ \text{apellido GÓMEZ} \end{array} \right] \end{array} \right]$$

$$(e) \left[ \begin{array}{l} \text{nombre HERMINIA} \\ \text{apellido } \boxed{1} \text{ GARCÍA} \\ \text{hermano } \left[ \begin{array}{l} \text{nombre CARLOS} \\ \text{apellido } \boxed{1} \end{array} \right] \end{array} \right]$$

$$(f) \left[ \begin{array}{l} \text{nombre HERMINIA} \\ \text{apellido GARCÍA} \end{array} \right]$$

$$(g) \left[ \begin{array}{l} \text{nombre CARLOS} \\ \text{apellido GARCÍA} \end{array} \right]$$

## Referencias

Bird, S., Klein, E., y Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. “O’Reilly Media, Inc.”.

- Blevins, J. P. (2011). Feature-based grammar. En *Non-Transformational Syntax*, pp. 297–324. Wiley Blackwell, Massachusetts.
- Gazdar, G. (1982). Phrase Structure Grammar. En Jacobson, P. y Pullum, G., editores, *The Nature of Syntactic Representation*, pp. 131–186. Reidel, Dordrecht.
- Moreno Sandoval, A. (2001). *Gramáticas de Unificación y Rasgos*. Antonio Machado, Madrid.
- Solias Arís, M. T. (2015). *Métodos formales en Lingüística*. Síntesis.