

# Gramáticas formales

Escuela Nacional de Antropología e Historia (ENAH)

Agosto – diciembre de 2015

# Antecedentes

- Lingüística matemática y computacional

# Alfabeto

- Conjunto de símbolos, finito y no vacío, que se representa con la letra griega  $\Sigma$
- a) Un alfabeto binario:  $\Sigma = \{0, 1\}$ .
- b) El conjunto de todas la letras minúsculas:  $\Sigma = \{a, b, \dots, z\}$ .

# Cadena

- Secuencia finita de símbolos seleccionados de un alfabeto
- 01101 es una cadena obtenida del alfabeto binario
- Cadena vacía.- contiene cero símbolos, y es representada comúnmente con los símbolos  $\epsilon$  o  $\epsilon$ .

# Cadena

- Conjunto de todas las cadenas obtenidas de un alfabeto se representa como  $\Sigma^*$
- $\{0, 1\}^* = \{e, 0, 1, 00, 01, 10, 11, 000, \dots\}$
- Concatenación.- permite obtener una cadena formada por la yuxtaposición de las cadenas originales
- Por ejemplo, si se tiene la cadena  $x = 01101$  y la cadena  $y = 110$ , su concatenación sería  $xy = 01101110$

# Lenguaje formal

- Conjunto de cadenas que pertenecen al conjunto de todas las cadenas posibles generadas de un alfabeto ( $\Sigma^*$ )
- De una manera simple se puede ver a toda lengua natural como un conjunto de cadenas (Hopcroft, Motwani y Ullman, 2001, pág. 30)

# Gramática formal

- Se debe tener una gramática que describa de manera precisa el conjunto de cadenas de un lenguaje
- Vocabulario terminal ( $V_T$ )
- Vocabulario no terminal ( $V_N$ )
- Estos dos vocabularios no comparten elementos entre sí y a la unión de ambos se le llama vocabulario ( $V$ )

# Gramática formal

- Reglas gramaticales.- conjunto finito de reglas de la forma abreviada  $\phi \rightarrow \psi$
- Se leen como 'ϕ es reescrito como ψ'
- Un sólo axioma  $S$  de *sentence*

# Gramática formal

- Así, se puede definir formalmente una gramática  $G$  como un conjunto de cuatro elementos  $(V_N, V_T, P, S)$

	Símbolos individuales (primeras letras del alfabeto)	Cadenas (últimas letras del alfabeto)
No terminal	$A, B, C, \dots$	$\dots, X, Y, Z$
Terminal	$a, b, c, \dots$	$\dots, x, y, z$
No especificado	$\alpha, \beta, \gamma, \dots$	$\dots, \chi, \psi, \omega$

# Gramática formal

- Derivación.- sucede cuando todas las cadenas son obtenidas de alguna regla
- Derivación terminada.- si y solo si la última cadena no puede ser reescrita por ninguna regla de la gramática
- Cadena terminal.- es generada por una derivación terminal que comienza con el axioma  $S$  y debe estar formada por símbolos terminales ( $x \in V_T^*$ )

# Gramática formal

- Lenguaje asociado a una gramática  $L(G)$ .- es el conjunto de cadenas terminales generadas por la gramática

# Gramática formal

$$G = (V_N, V_T, \{S\}, P)$$

$$V_N = \{S, A, B, C\}$$

$$V_T = \{a, b, c\}$$

$$P = \{1. S \rightarrow ABC$$

$$2. A \rightarrow aA$$

$$3. A \rightarrow a$$

$$4. B \rightarrow Bb$$

$$5. B \rightarrow b$$

$$6. BC \rightarrow Bcc$$

$$7. ab \rightarrow ba\}$$

# Gramática formal

- Dada la cadena:  $BCA$ ,  $BccA$ ,  $BbccA$ ,  $BbcccaA$
- ¿Es una derivación?
- ¿Es una derivación terminada?
- ¿Es una cadena terminal?

# Gramática formal

- Dada la cadena: *BCA, BccA, BbccA, BbccA, BbccA, bbccA*
- ¿Es una derivación?
- ¿Es una derivación terminada?
- ¿Es una cadena terminal?

# Gramática formal

- Dada la cadena:  $S, ABC, aBC, aBcc, abcc, bacc$
- ¿Es una derivación?
- ¿Es una derivación terminada?
- ¿Es una cadena terminal?

# Tipos de gramáticas formales

- Tipo 0. Llamadas sistemas de reescritura no restringida ya que no tienen ninguna restricción en sus reglas.

# Tipos de gramáticas formales

- Tipo 1. Llamadas gramáticas sensibles al contexto. Cada regla es de la forma  $\varphi A \psi \rightarrow \varphi \omega \psi$ , donde  $\varphi$  y  $\psi$  podrían ser nulas.

La cadena  $\omega$  no puede ser nula.

En cada regla un solo símbolo no terminal ( $A$ ) es reescrito como una cadena no nula.

$A$  se reescribe como  $\omega$  cuando aparece con la cadena  $\varphi$  a su izquierda y la cadena  $\psi$  a su derecha.

En lingüística, este tipo de reglas se expresan como  $A \rightarrow \omega / \varphi \_ \psi$ .

# Tipos de gramáticas formales

- Tipo 2. Llamadas gramáticas libres de contexto. Cada regla es de la forma  $A \rightarrow \omega$ .

Donde  $\omega$  no puede ser nula ( $\omega \neq e$ ).

Las cadenas  $\varphi$  y  $\psi$  son nulas.

El símbolo  $A$  puede ser reescrito como  $\omega$  sin importar el contexto en el que aparece; de allí el nombre del tipo de gramática.

# Tipos de gramáticas formales

- Tipo 3. Llamadas gramáticas regulares o de estados finitos. Cada regla es de la forma  $A \rightarrow xB$  o de la forma  $A \rightarrow x$ .

Donde  $x$  no puede ser nula ( $x \neq e$ ).

Este tipo de gramática agrega la restricción de que el lado derecho de cada regla debe ser una cadena formada de símbolos terminales seguida de, a lo mucho, un símbolo no terminal.

# Tipos de lenguajes

- Los lenguajes producidos por cada tipo de gramática reciben su nombre de acuerdo con el tipo que los genera:
  - Tipo 0, llamados conjuntos recursivamente enumerables.
  - Tipo 1, llamados lenguajes sensibles al contexto.
  - Tipo 2, llamados lenguajes libres de contexto.
  - Tipo 3, llamados lenguajes regulares, conjuntos regulares o lenguajes de estados finitos.

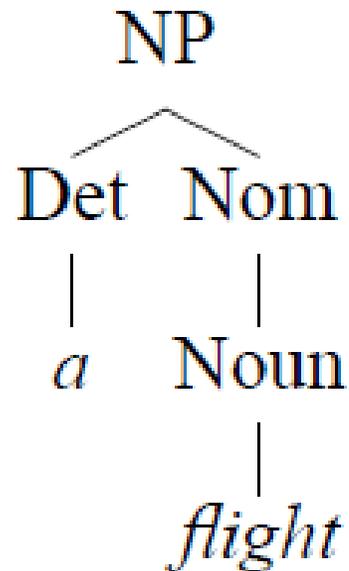
# Context Free Grammar (CFG)

## Noun phrase (NP)

- $NP \rightarrow Det\ Nominal$
- $NP \rightarrow ProperNoun$
- $Nominal \rightarrow Noun \mid Nominal\ Noun$
- $Det \rightarrow a$
- $Det \rightarrow the$
- $Noun \rightarrow flight$

# Context Free Grammar (CFG)

- Terminal symbols: lexicon
- Non-terminal symbols: part-of-speech
- Generate or recognize:
- *NP*
- *Det Nominal*
- *Det Noun*
- *a flight*



Fin