Herramientas para el procesamiento de textos en Python

Profesor Titular
Martín Kondratzky
Colaboradores
Fernando Carranza, Macarena Fernández Urquiza, Fernando Schiaffino
Colaboradores invitados
Julia Milanese, Federico Alvarez
Catalina Rubio
Victoria Colombo

Sábado 19/05/2018

Esta unidad

Número de clase: Clases 7 y 8
Contenidos de la parte teórica

- Gramáticas formales
- Jerarquía de gramáticas formales de Chomsky
- Tipos de gramáticas: gramáticas basadas en constituyentes, gramáticas de dependencias, gramáticas categoriales

La estructura sintáctica y su naturaleza jerárquica

La tradición lingüística acuerda respecto de que las oraciones de las lenguas naturales están organizadas jerárquicamente:

- Existen operaciones sintácticas que aplican a grupos de palabras, mostrando que hay palabras más "cercanas" que otras
  - Coordinación
  - Interrogación
  - Elipsis
  - Focalización y topicalización.
  - Pronominalización.
  - Alteración del orden de palabras
- Los hablantes son sensibles a esas agrupaciones de palabras antes que a cuestiones de linealidad.
  - En la adquisición, los chicos no cometen errores del tipo "mover a la posición inicial el primer verbo".
  - Las alternancias argumentales muestran que los hablantes interpretan los argumentos en función de la estructura en la que se encuentran y no en términos de cuál aparece primero.

Las formas de representar la naturaleza jerárquica

En la primera clase examinamos distintos tipos de lenguajes entendidos como conjuntos de cadenas. Estos lenguajes pueden ser caracterizados por intensión por distintos mecanismos. La posibilidad de generar lenguajes de estos mecanismos es lo que se conoce como su **generación débil**.

Las formas de representar la naturaleza jerárquica

En la primera clase examinamos distintos tipos de lenguajes entendidos como conjuntos de cadenas. Estos lenguajes pueden ser caracterizados por intensión por distintos mecanismos. La posibilidad de generar lenguajes de estos mecanismos es lo que se conoce como su **generación débil**.
Si dos objetos matemáticos distintos generan el mismo lenguaje, se dice que ambos son **débilmente equivalentes**

Las gramáticas son uno de estos mecanismos que permiten generar lenguajes, pero poseen la característica adicional de que además de permitir responder a la pregunta de si una cadena pertenece o no a un lenguaje, también sirven para asignar estructura a las cadenas.

Las gramáticas son uno de estos mecanismos que permiten generar lenguajes, pero poseen la característica adicional de que además de permitir responder a la pregunta de si una cadena pertenece o no a un lenguaje, también sirven para asignar estructura a las cadenas.
La capacidad de una gramática de generar una estructura es lo que se conoce como su **capacidad generativa fuerte**.

Las gramáticas

A grammar (...) is some explicit device form (...) selecting a subset of strings that are grammatical, from the set of all possible strings formed from an initially given language.
(Partee : 433)

Lenguajes Tipo 0	Gramáticas irrestrictas
Lenguajes tipo 1	Gramáticas sensibles al contexto
Lenguajes tipo 2	Gramáticas independientes de contexto
Lenguajes tipo 3	Gramáticas regulares

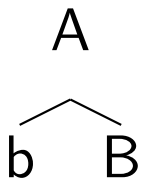
Las gramáticas se expresan usualmente en forma de reglas de reescritura de la forma $X \rightarrow Z$ (X se reescribe como Z) o de grafos dirigidos, coloquialmente denominados “árboles”.

Gramáticas Regulares

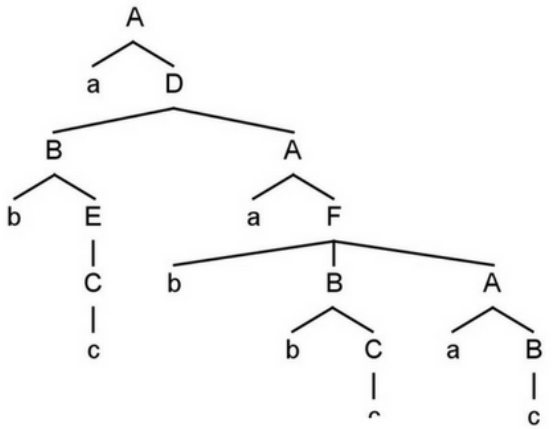
Las gramáticas regulares son aquellas que solo pueden producir lenguajes regulares. Estas gramáticas restringen sus reglas de reescritura a solamente dos tipos:

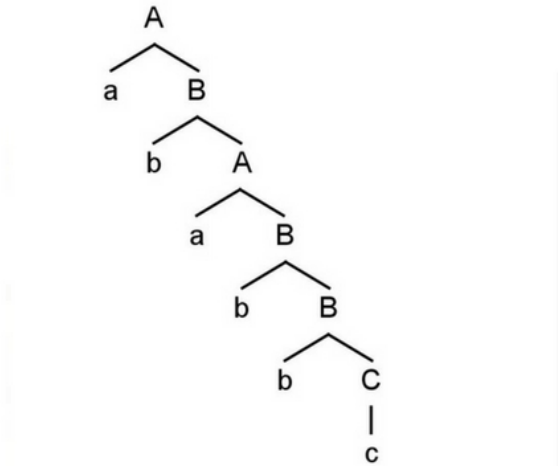
- $A \rightarrow b\ B$
- $A \rightarrow b$

Es decir, a dos clases de árboles:



Gramáticas Regulares



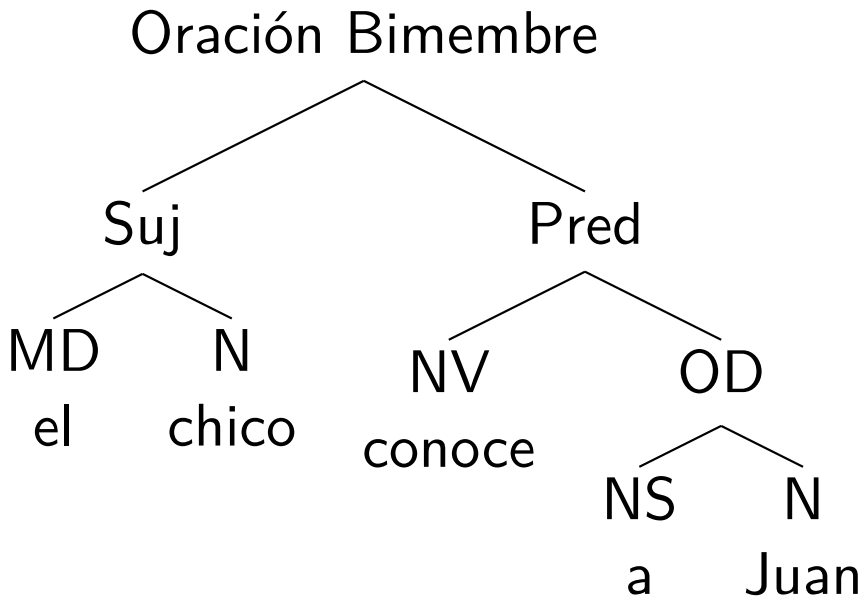


Gramáticas Regulares

- Los gramáticas regulares son débilmente equivalentes a las expresiones regulares (i.e. generan y reconocen exactamente los mismos lenguajes).
- Todo lenguaje finito construido a partir de un alfabeto finito es necesariamente un lenguaje regular.
- Usualmente, para cualquier operación que involucre lenguajes regulares se utilizan expresiones regulares antes que gramáticas regulares.

Gramáticas Regulares

Si se compara la estructura de las lenguas naturales con la de las gramáticas regulares, resulta obvio que estas gramáticas son incapaces de arrojar un análisis sintáctico de las lenguas naturales, incluso si uno recurre a un análisis escolar estándar:



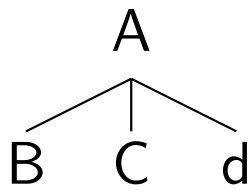
Gramáticas Regulares

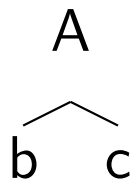
Por esta razón, los parsers pensados para lenguaje natural nunca están basados en esta clase de gramáticas.

Gramáticas Independientes de Contexto

Las gramáticas independientes de contexto son capaces de generar lenguajes regulares e independientes de contexto y restringen sus reglas de reescritura a reglas que del lado izquierdo tengan un solo elemento. No tienen ninguna restricción respecto de la clase de cosas que puede haber del lado derecho.

$A \rightarrow$ lo que quieras





Gramáticas Independientes de Contexto

El principal fenómeno que queda por fuera de las posibilidades de las gramáticas independientes de contexto son los constituyentes discontinuos. No obstante, como implementar una gramática que pueda dar cuenta de los constituyentes no discontinuos suele ser demasiado costoso, la mayoría de los parsers se conforma simplemente con las gramáticas independientes de contexto y aceptan que siempre quedará un margen de estructuras para las cuales no se dará un parseo satisfactorio.

Gramáticas Sensibles al Contexto

Las gramáticas sensibles al contexto generan lenguajes regulares, independientes de contexto y sensibles al contexto. Existen diferentes formas equivalentes de definir las gramáticas sensibles al contexto.

- Gramáticas cuyas reglas de reescritura permiten reescribir un no terminal en función de los elementos que lo rodean. ABC $\rightarrow$ ADC
- Gramáticas cuyas reglas de reescritura aceptan símbolos terminales o no terminales a su izquierda y pueden reescribir estos símbolos no terminales a la derecha pero no pueden borrar ningún símbolo.

Gramáticas Irrestrictas

Las gramáticas irrestrictas pueden generar todos los tipos de lenguajes de la jerarquía de Chomsky. Sus reglas de reescritura pueden tomar un conjunto elementos que incluyan al menos un símbolo no terminal y reescribirlo en otro conjunto de elementos.

Gramáticas que generan lenguajes independientes de contexto

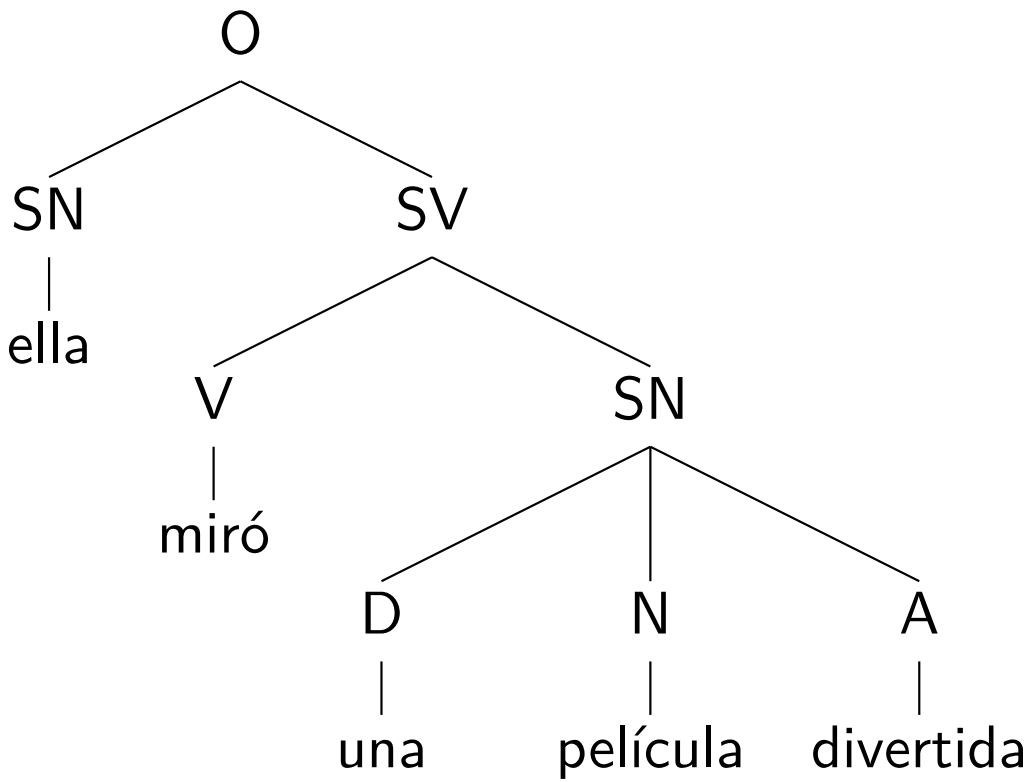
Si bien las gramáticas independientes de contexto son más restringidas que las gramáticas de las lenguas naturales, son gramáticas tratables y fáciles de implementar, por lo que los parsers suelen manejarse con este tipo de gramáticas. A continuación vamos a tratar de trabajar con ellas para poder ver después cómo implementarlas con distintos tipos de parsers.

Las formas de representar la naturaleza jerárquica

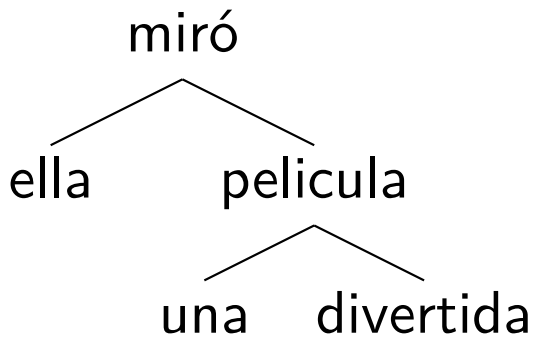
Sabemos que queremos trabajar con gramáticas que generen débilmente lenguajes independientes de contexto. Ahora bien, existen al menos tres implementaciones de gramáticas diferentes capaces de hacer eso. Las tres se distinguen según el modo en que conciben la naturaleza jerárquica de la sintaxis:

- La jerarquía está dada por agrupaciones de palabras llamadas constituyentes
- La jerarquía se construye como encadenamientos de inferencias lógicas a partir de relaciones función-argumento
- La jerarquía está dada a partir de relaciones de dependencia entre palabras.

La estructura sintáctica en términos de constituyentes



La estructura sintáctica en términos de dependencias



La estructura sintáctica en términos de funciones y argumentos

$$\dfrac{\dfrac{ella}{n}\ >\text{Lex} \qquad \dfrac{\dfrac{miró}{(s\backslash n)/n}\ >\text{Lex.} \quad \dfrac{\dfrac{una}{n/n}\ >\text{Lex.} \quad \dfrac{película}{n}\ >\text{Lex.}}{n}\ >\text{Apl.}}{s\backslash n}\ >\text{Apl.}}{s}\ >\text{Apl.}$$

Las Gramáticas de constituyentes

Las gramáticas de constituyentes suelen tener las siguientes características:

- Single root condition: Solo tienen un nodo inicial.
- Acíclicas: Generan grafos dirigidos acíclicos.
- Non Tangling condition: Si un nodo α precede a un nodo β, todos los elementos dominados por α deben preceder a los elementos dominados por β

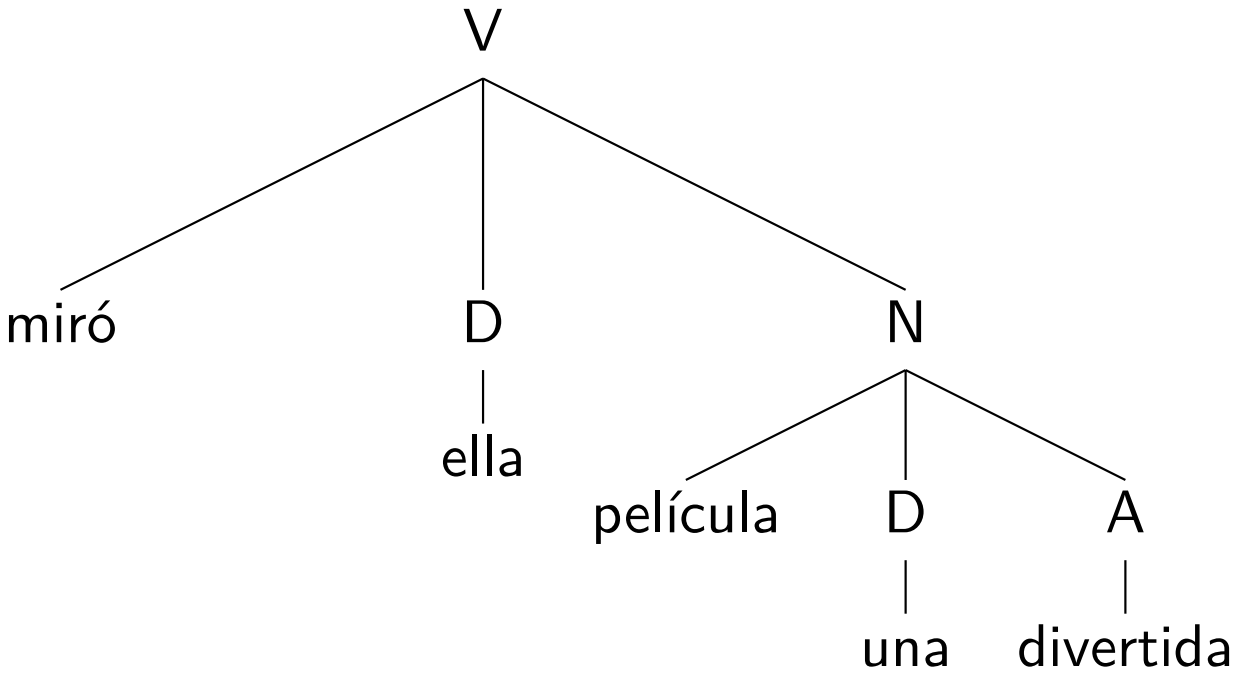
Las gramáticas de dependencias

Las gramáticas de dependencias suelen tener las siguientes características:

- Conectividad: Para toda palabra de la cadena debe existir una relación de dependencia con otra.
- Aciclicidad: El nodo inicial no puede ser regido por otro nodo.
- Proyectividad: Cada relación de dependencia debe ser grafo adyacente, es decir, entre dos elementos α y β que establezcan relación de dependencia, solo puede haber elementos que o bien dependan de α o bien dependan de β.

Gramáticas de dependencias con categorías

Existen algunas gramáticas de dependencias que agregan categorías.



Gramáticas de dependencias con categorías

Formalmente, estas gramáticas se asemejan a gramáticas expresadas en Greibach Normal Form:

- Gramáticas en Greibach Normal Form: Cada regla de reescritura tiene un solo no terminal del lado izquierdo y el lado derecho debe comenzar obligatoriamente con un terminal.
- Gramáticas de dependencias con categorías: Cada regla de reescritura tiene un solo no terminal del lado izquierdo y tiene un terminal del lado derecho seguido de no terminales.

Las gramáticas categoriales conciben las estructuras a partir de funciones y argumentos. En notación de Steedman, las funciones tienen la forma siguiente:

- X/Z: Una categoría que toma Z como argumento a su derecha y devuelve X
- X\Z: Una categoría que toma Z como argumento a su izquierda y devuelve X.

Gramáticas categoriales

El sistema de Lambek es la versión axiomatizada más clásica de la Gramática Categorial. Este sistema consta de los siguientes axiomas:

- $x \rightarrow x$
- - $(xy)z \rightarrow x(yz)$
  - $x(yz) \rightarrow (xy)z$

Y las reglas de inferencia son:

- si $xy \rightarrow z$, entonces $x \rightarrow z/y$
  si $xy \rightarrow z$ entonces $y \rightarrow z\backslash x$
- si $x \rightarrow z/y$ entonces $xy \rightarrow z$
  si $y \rightarrow z\backslash x$ entonces $xy \rightarrow z$
- si $x \rightarrow y \wedge y \rightarrow z$, entonces $x \rightarrow z$.

Gramáticas Categoriales

Diferentes marcos categoriales asumen conjuntos distintos de reglas. Las siguientes son las reglas que asume el parser OpenCCG, cuya implementación vamos a explorar luego.

- **Aplicación Funcional a derecha (binaria)** $(>)$:
  $X/Y_S\ Y \rightarrow X$
- **Aplicación Funcional a izquierda (binaria)** $(<)$
  $Y\ X\backslash Y_S \rightarrow X$
- **Composición (binaria) a derecha (>B)**:
  $X/Y_S\ Y/Z_S \rightarrow X/Z_S$
- **Composición (binaria) a izquierda (<B)**:
  $Y\backslash Z_S\ X\backslash Y_S \rightarrow X\backslash Z_S$
- **Raising (unaria) (>T)**:
  $X \rightarrow Y/(Y\backslash X)_S$
- **Raising (unaria) (<T)**:
  $X \rightarrow Y\backslash(Y/X)_S$

La diferencia más importante en relación con las gramáticas basadas en constituyentes y las gramáticas basadas en dependencias es que, en la arquitectura de las gramáticas categoriales, las reglas son un número reducido y están hardcodeadas. La programación de una gramática categorial consiste principalmente en la construcción de un léxico.