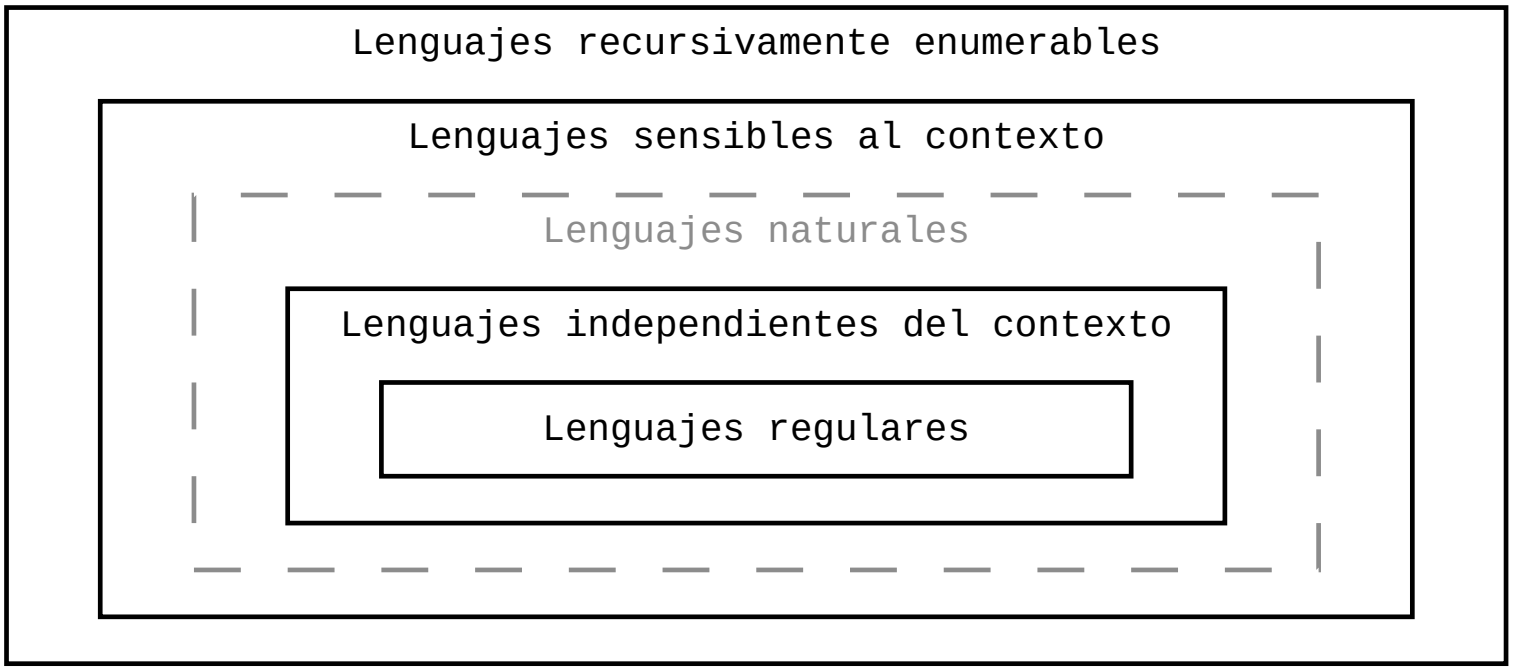
Índice

Análisis Sintáctico

- a.k.a. *[Syntactic] Parsing*
- En base a la gramática de ese lenguaje.
- **Objetivos:**
  - Comprobar que el texto analizado es sintácticamente correcto (reconocedores o *scanners*).
  - Obtener su estructura sintáctica (analizadores o *parsers*).
- **Aplicaciones:**
  - Corrección gramatical
  - Interpretación semántica
  - Sistemas de diálogo
  - Traducción automática
  - *Text Mining*

Consideraciones

1. **Expresividad de la gramática:** formalismos gramaticales de expresividad y complejidad de análisis crecientes



Jerarquía de Chomsky

Consideraciones (cont.)

2. **Cobertura del lenguaje:**
   - Sistemas de cobertura amplia:
     - Capaces de tratar cualquier texto no restringido.
     - Gramáticas muy complejas.
   - Sistemas aplicados a sublenguajes (contextos restringidos): mayor sencillez

3. **Estrategia de análisis:**
   - Análisis descendente (*top-down*), ascendente (*bottom-up*) o híbrido.
   - Dirección del análisis: de izqda. a dcha. (tb. viceversa) o bidireccional.

4. **Gestión de la ambigüedad y el no determinismo:**
   - Ambigüedad: cuando existe más de una estructura sintáctica posible para todo o parte del texto analizado.
   - No determinismo: cuando durante el proceso de análisis es posible tomar varios caminos diferentes (cuando surge ambigüedad).

Evolución de los Analizadores

- Dos corrientes principales:
  1. Uso de técnicas de análisis sintáctico derivadas de las utilizadas en análisis de lenguajes formales y construcción de compiladores.

     Ej: Extensiones *LL(k)* y *LR(k)*, algoritmo *CYK*, algoritmo *de Earley*.
  2. Uso de técnicas derivadas de la Inteligencia Artificial.

Recursos Lingüísticos para *Parsing*

1. **Gramática** G que define el lenguaje:

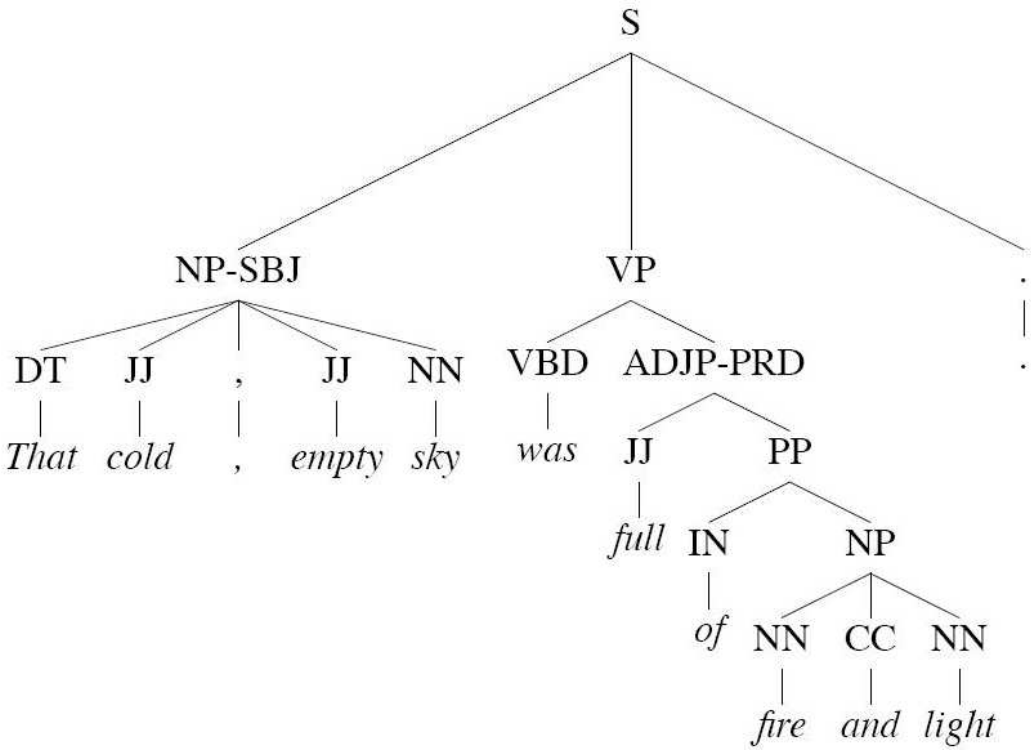
$$G = \{N, \Sigma, P, S\} \quad \textit{donde} \quad \begin{array}{l} N \equiv \text{conjunto de símbolos no terminales} \\ \Sigma \equiv \text{conjunto de símbolos terminales} \\ P \equiv \text{conjunto de reglas} \\ S \equiv \text{símbolo raíz} \end{array}$$

Ej. Gramática (sencilla) oraciones SUJ-V-OBJ:

$$\begin{array}{lcl} S & \rightarrow & NP \quad VP \\ NP & \rightarrow & det \quad nombre \\ VP & \rightarrow & verbo \quad NP \end{array}$$

Recursos Lingüísticos para *Parsing* (cont.)

2. **Banco de árboles (treebank).** Texto donde no sólo cada palabra está acompañada de su etiqueta correcta, sino que además cada frase aparece completamente analizada sintácticamente (notación parentizada). A partir de él se puede:
   - Extraer [parte de] las reglas de la gramática.
   - Probabilidad de uso asociada a cada una.




```
((S
   (NP-SBJ (DT That)
     (JJ cold) (, ,)
     (JJ empty) (NN sky) )
   (VP (VBD was)
     (ADJP-PRD (JJ full)
       (PP (IN of)
         (NP (NN fire)
           (CC and)
           (NN light) ))))
   (. .) ))
```

Análisis Sintáctico Superficial

- Problemas del **análisis sintáctico completo/clásico** (*full parsing*):
  - Requiere conocimiento/recursos lingüísticos complejos (gramáticas, *treebanks*)
  - Escasa cobertura de las gramáticas
  - Escasa robustez
  - Alto coste
- Sin embargo no siempre es necesario que dicha información sea completa/exhaustiva pues sólo nos interesan ciertas estructuras o relaciones. P.ej.:
  - IR: más centrado en frases nominales
  - IE: sólo los segmentos de texto con información relevante

Análisis Sintáctico Superficial (cont.)

- Solución: **análisis sintáctico superficial** (*shallow parsing*; a.k.a. **chunking**, *partial parsing*):
  - Devuelve una representación *"superficial"* (i.e. aproximativa, incompleta) de la estructura sintáctica del texto:
    - Opera en base a **grupos de palabras** o **chunks**
    - Plana, i.e. no contempla estructuras arborescentes
  - Requerimientos menores
  - Mayor robustez
  - Bajo coste

Chunk

- **Def.:** grupo de palabras (segmento) que funcionan conjuntamente como un única *palabra con contenido*:
  - Nombre: funciona a modo de frase/grupo nominal (NP)
  - Adjetivo: a modo de frase/grupo adjetival (AP)
  - Verbo: a modo de frase/grupo verbal (VP)
  - Preposición*: a modo de frase/grupo preposicional (PP)
- Pero no son *frases* en el sentido estricto, sino aproximaciones.
- No hay estructuras recursivas (p.ej. *criador de caballos de carreras*).
  - Se simplifica el proceso de detección.
- (En inglés) Se devuelve el segmento desde la palabra inicial del grupo hasta el núcleo, desechando los modificadores posteriores
  - Influido por la sintaxis (en inglés los modificadores preceden al núcleo)
  - Se evita el problema de la ambigüedad en las adjunciones:

$[_{VP}$ *vi*$]$ $[_{PP}$ *a*$]$ $[_{NP}$ *un hombre*$]$ $[_{PP}$ *en*$]$ $[_{NP}$ *una colina*$]$ $[_{PP}$ *con*$]$ $[_{NP}$ *un telescopio*$]$

Como Etiquetación de Palabras

- Un proceso de *chunking* implica:
  - Localizar el segmento/grupo de palabras
  - Identificar su clase
- Puede verse como un **proceso de etiquetación**. Dos enfoques posibles:
  (1) Como **etiquetación de palabras** (*IOB tagging*).
  (2) Como **etiquetación de separaciones entre palabras** (parentización).

Como Etiquetación de Palabras

Consiste en **identificar las palabras que integran el chunk** (*IOB tagging*):

- Se etiquetan las **palabras**.
- Las **chunk tags** indican dónde comienza un nuevo *chunk*, qué palabras contiene (más sencillo que detectar dónde termina) y el tipo del *chunk* (*tagset* ampliable según categorías consideradas: NP, VP, PP ...):
  - B (*Beginning*): si es la palabra inicial del *chunk*
  - I (*Internal*): si está en el interior del *chunk*
  - O (*Outside*): si está fuera del *chunk*

The	morning	flight	from	Denver	has	arrived
B_NP	I_NP	I_NP	B_PP	B_NP	B_VP	I_VP
B_NP	I_NP	I_NP	O	B_NP	O	O

[$_{NP}$ The morning flight] from [$_{NP}$ Denver] has arrived.

Como Etiquetación de Separaciones entre Palabras

Consiste en **delimitar el chunk mediante paréntesis** (parentización):

- Se etiquetan las **separaciones entre palabras**
- Las **gap tags** indican los límites y clase del *chunk* (*tagset* ampliable según categorías consideradas)

Beginning	End	Between	No bracket (outside)	No bracket (inside)
[*NP*	*NP*]	*NP*] [*NP*	*Out*	*In*

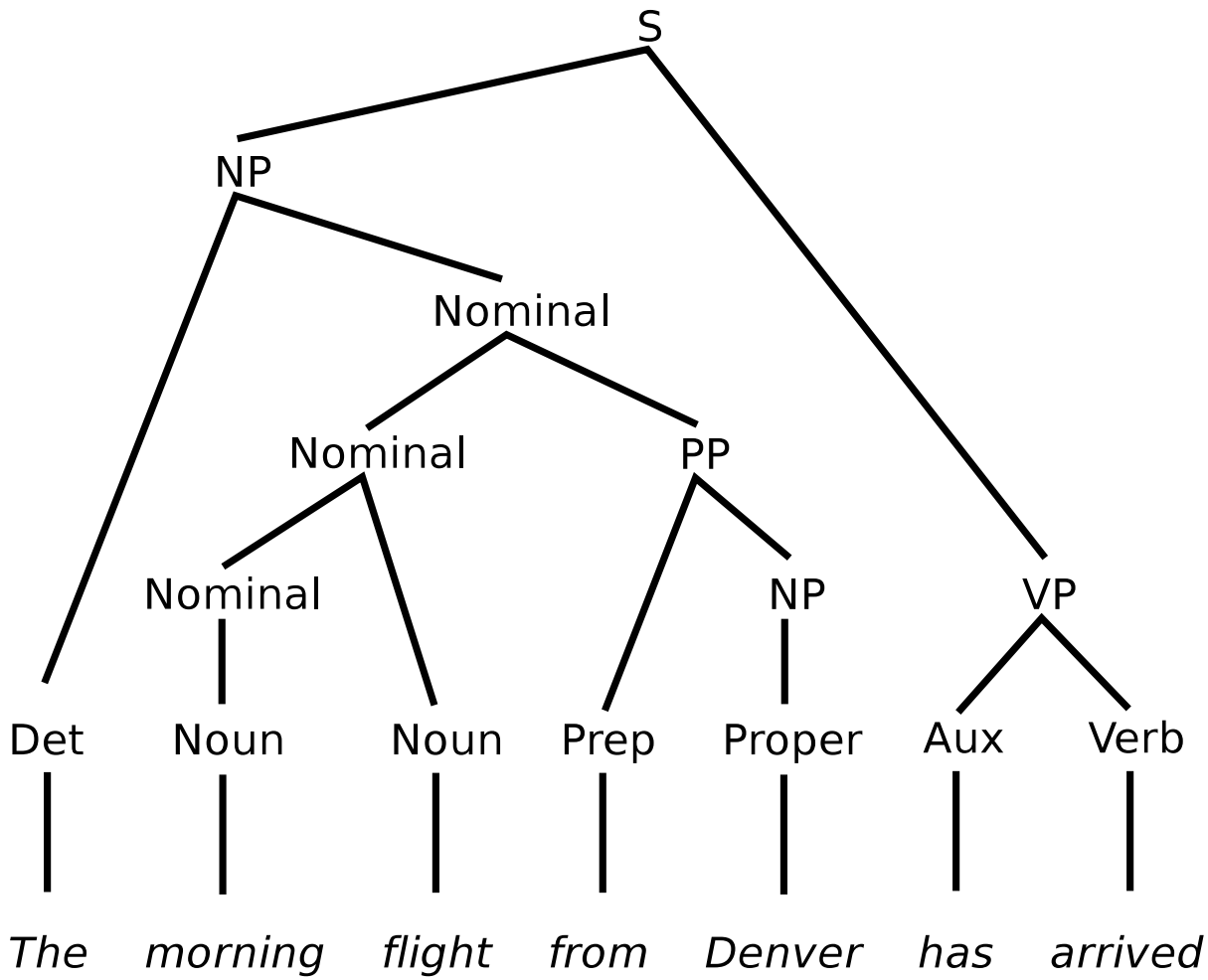
[$_{NP}$ The $_{In}$ morning $_{In}$ flight $_{NP}$] from [$_{NP}$ Denver $_{NP}$] has $_{Out}$ arrived.

[$_{NP}$ The morning flight] from [$_{NP}$ Denver] has arrived.

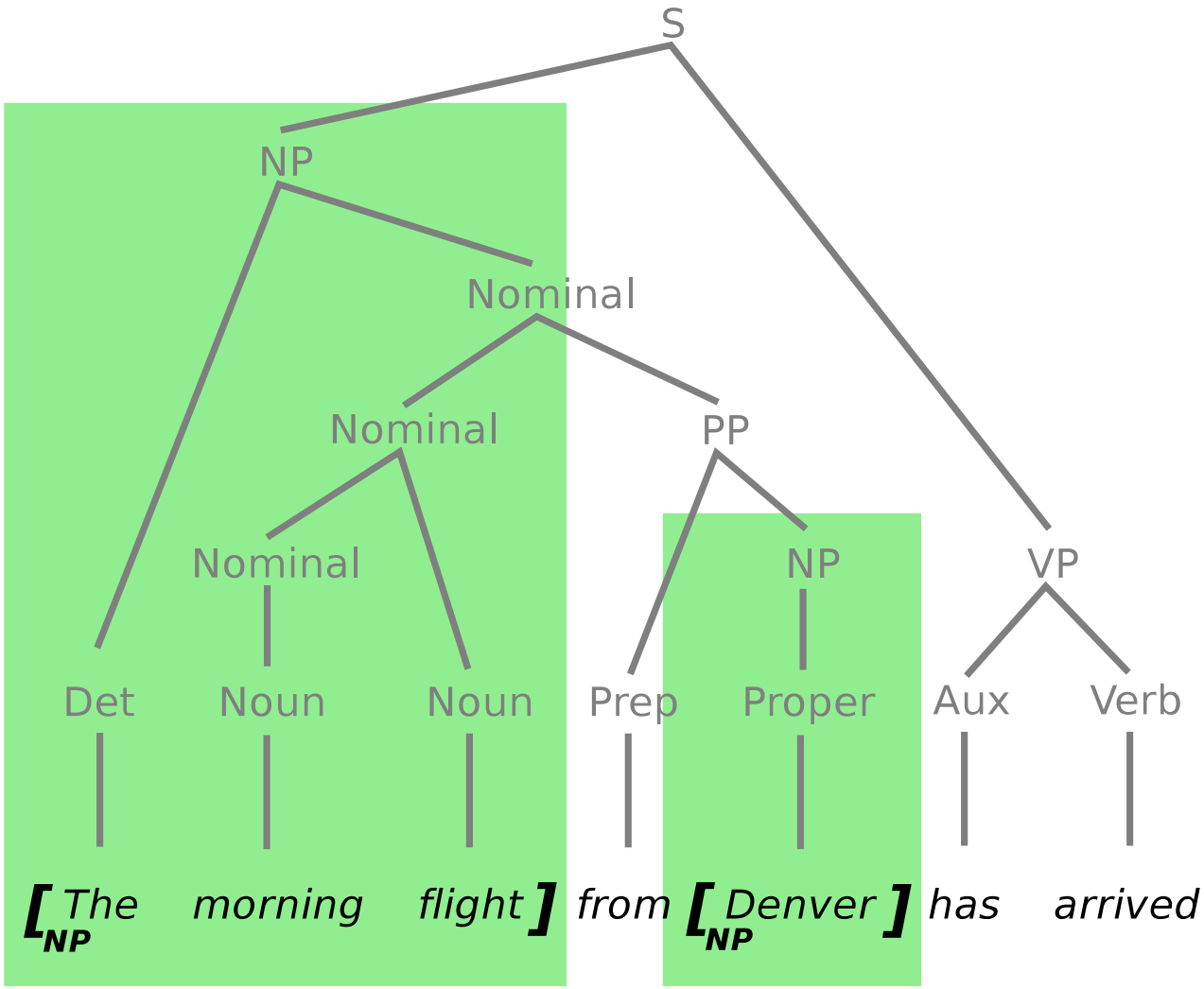
Ejemplo

The morning flight from Denver has arrived

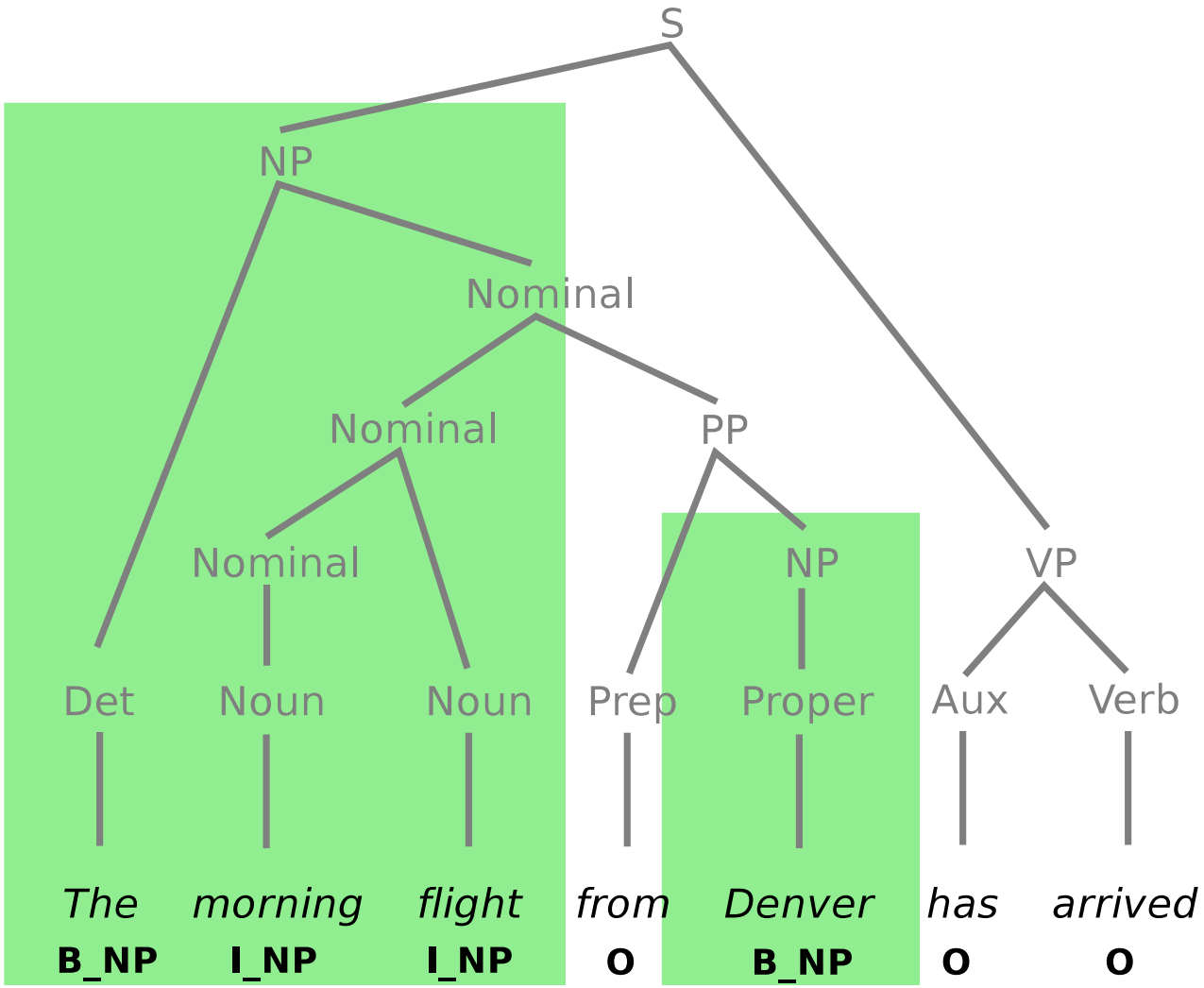
Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
The
morning
flight
from
Denver
has
arrived

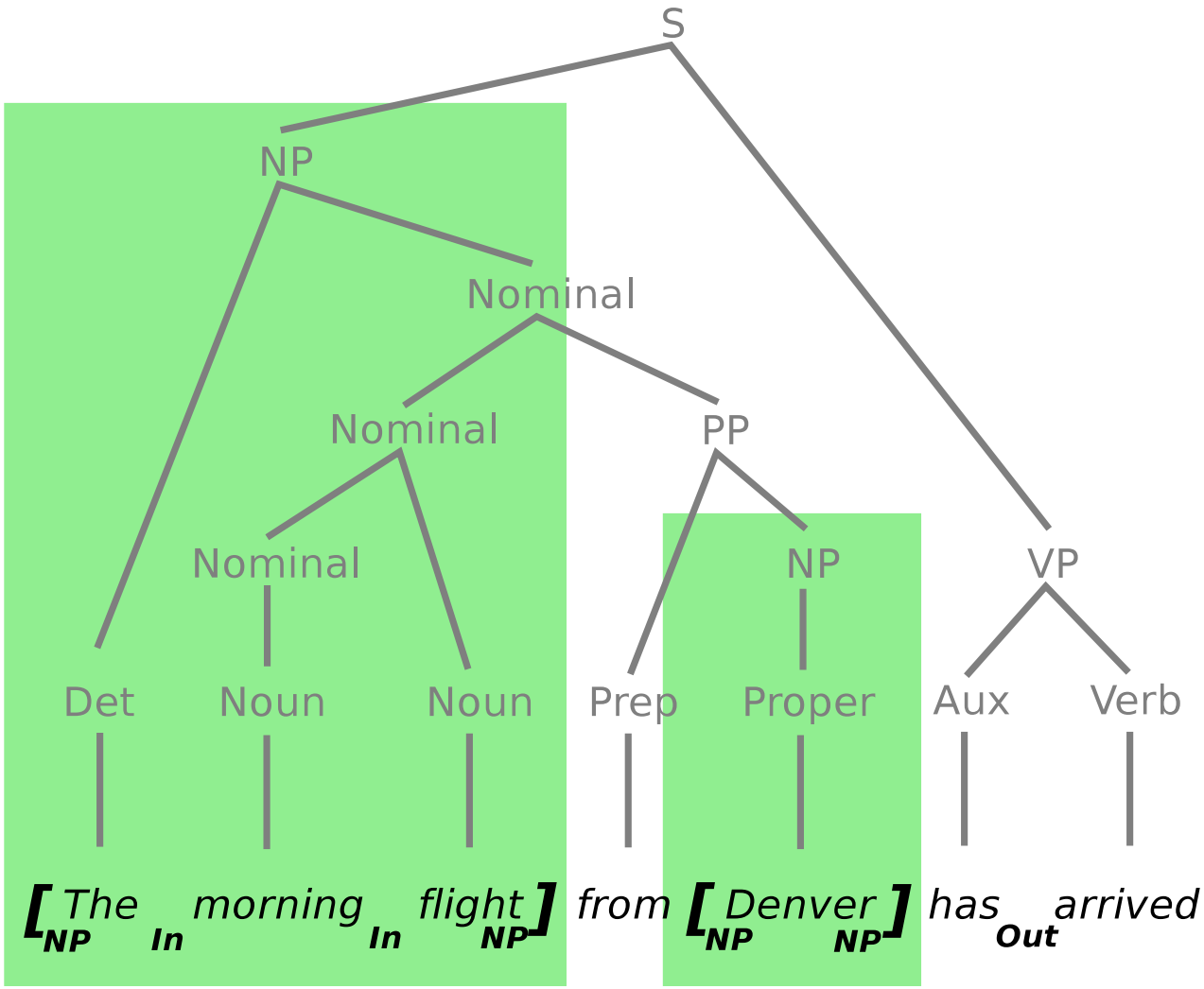
Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
[NP The morning flight] from [NP Denver] has arrived

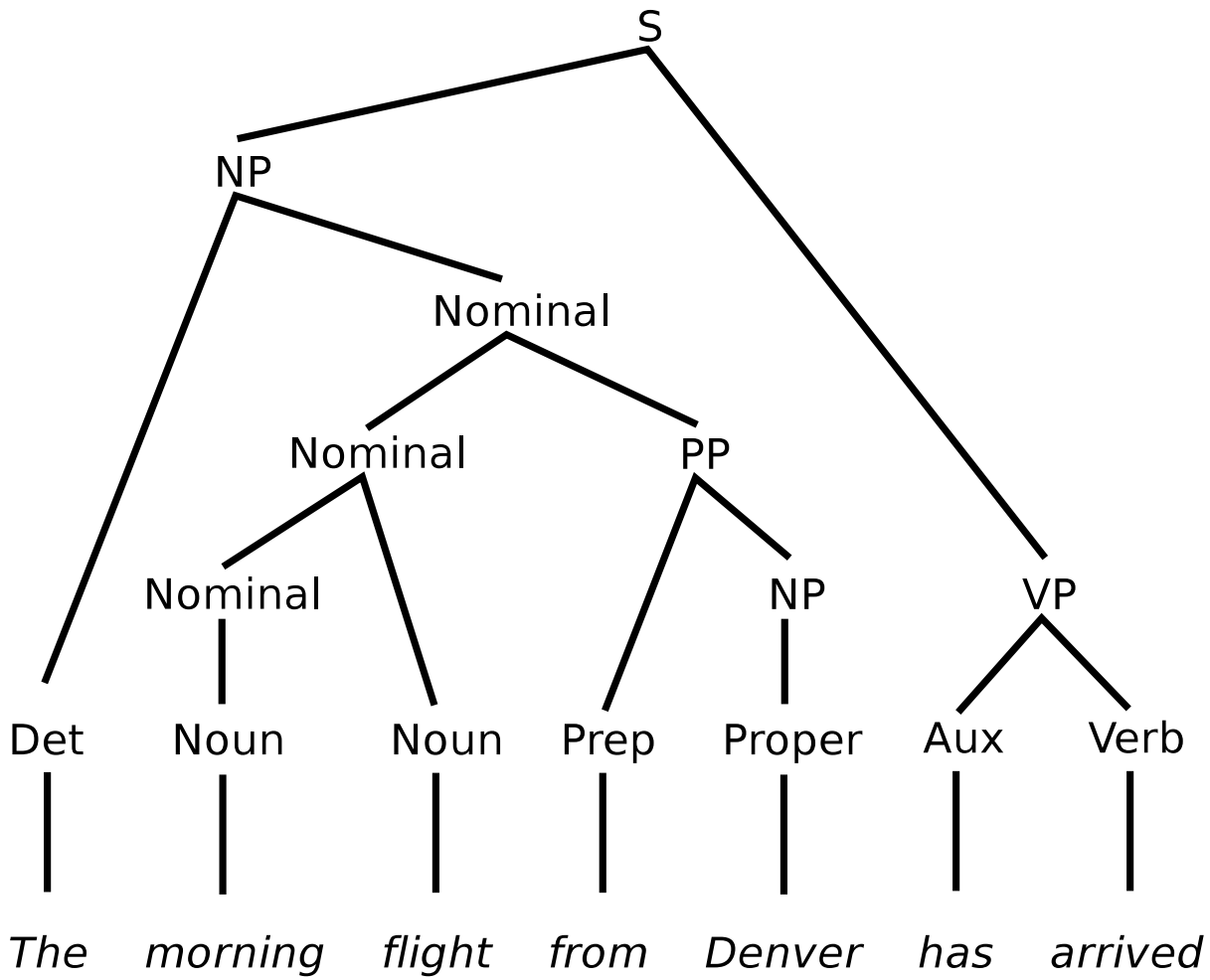
Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
The
morning
flight
from
Denver
has
arrived
B_NP
I_NP
I_NP
O
B_NP
O
O

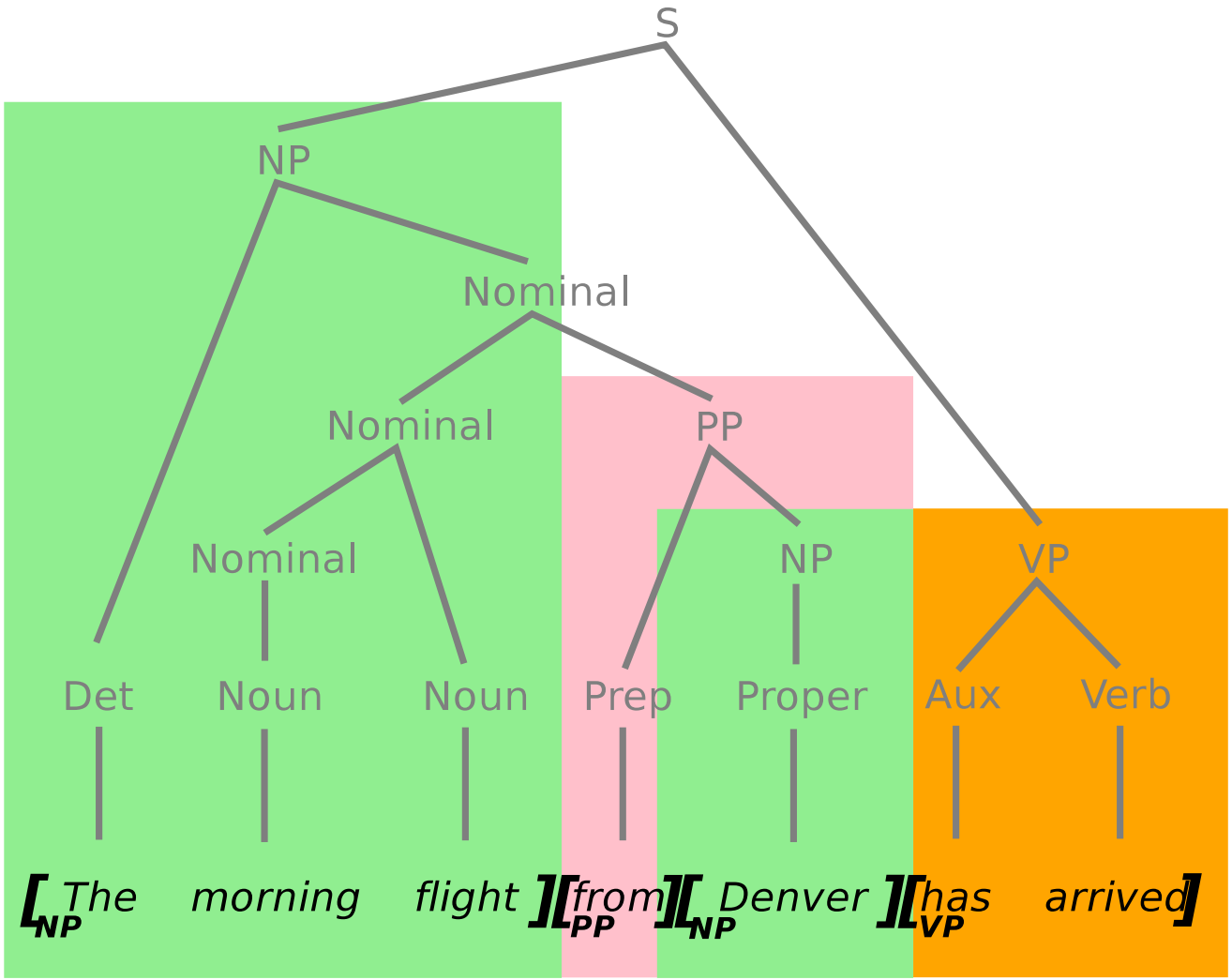
Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
[The morning flight]NP from [Denver]NP has arrived
In In NP NP Out

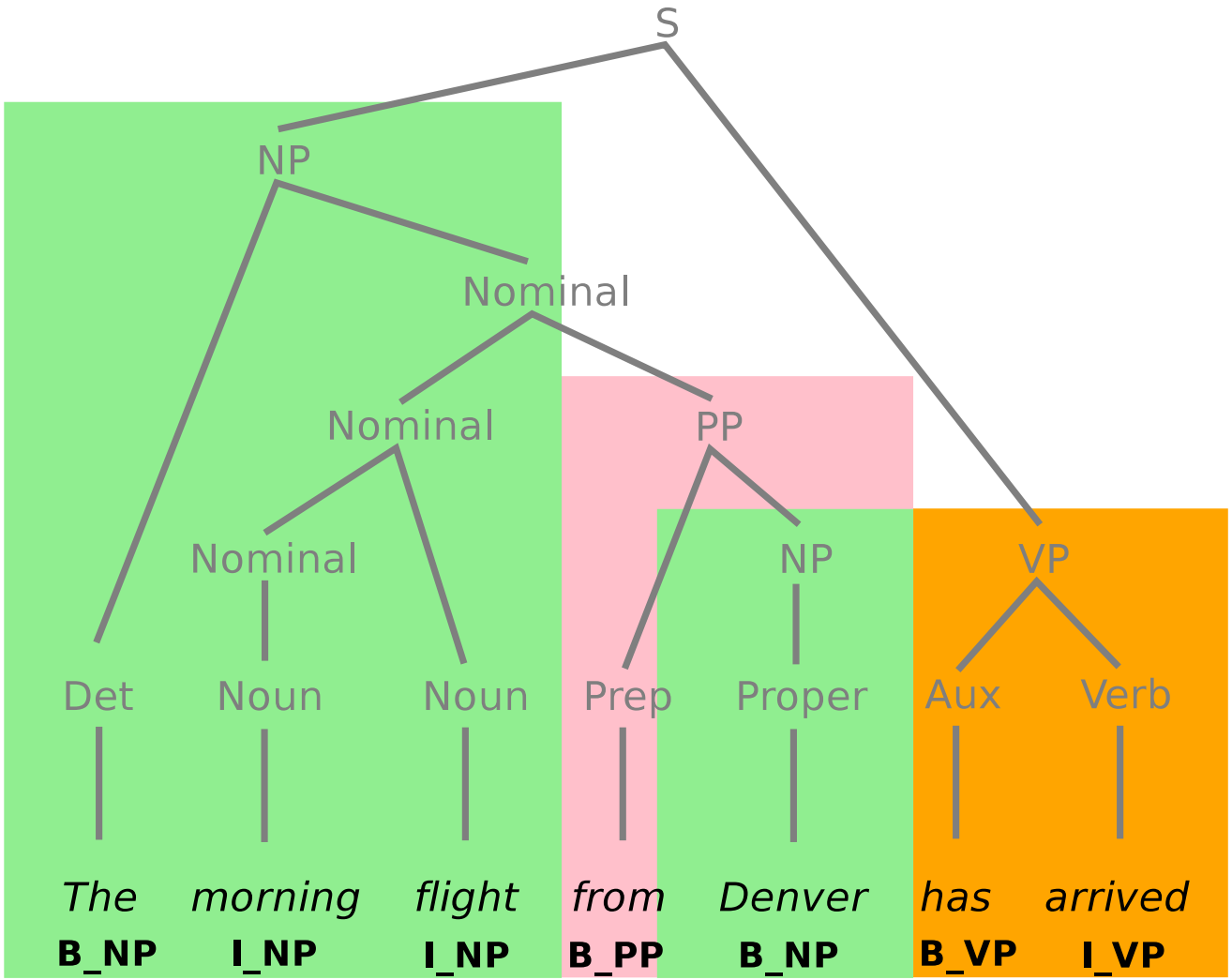
Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
The
morning
flight
from
Denver
has
arrived

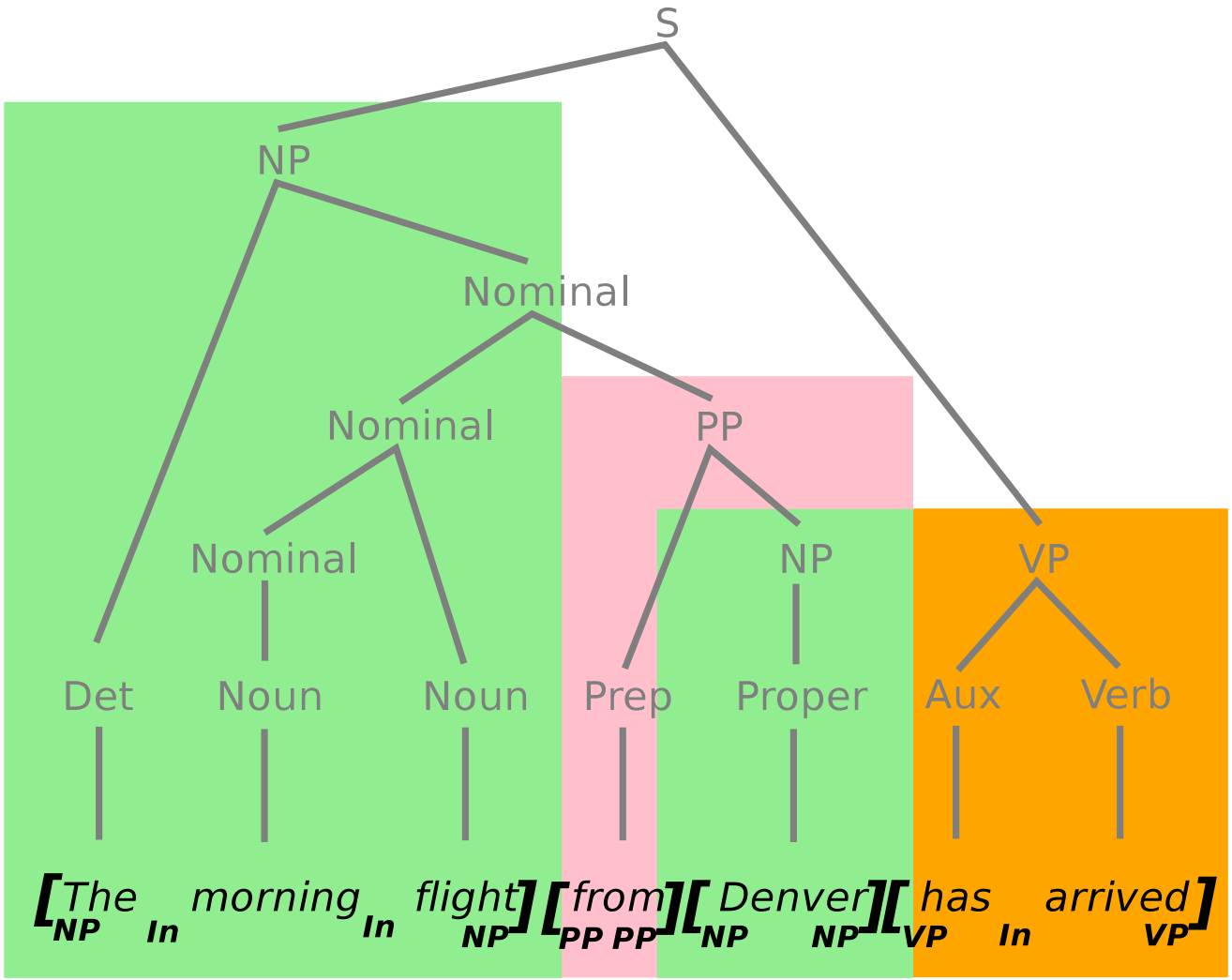
Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
[NP The morning flight][PP from][NP Denver][VP has arrived]

Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
The
morning
flight
from
Denver
has
arrived
B_NP
I_NP
I_NP
B_PP
B_NP
B_VP
I_VP

Ejemplo


S
NP
Nominal
Nominal
PP
Nominal
NP
VP
Det
Noun
Noun
Prep
Proper
Aux
Verb
[The morning flight][from][Denver][has arrived]
NP In In NP PP PP NP NP VP In VP

Implementación del Proceso de Chunking

Tres enfoques:

(1) Mediante correspondencia de patrones.

(2) Mediante reglas [aprendidas automáticamente]

(3) Mediante clasificadores secuenciales (ej. HMM)

Mediante Correspondencia de Patrones

- a.k.a. *finite-state role-based chunking*
- Se generan manualmente **patrones** que capturen las estructuras de interés:
  - Definidos en base a etiquetas, palabras, lemas, etc.
  - De izqda. a drcha.
  - *Longest matching*
  - No superposición, i.e. el siguiente *matching* empieza justo a continuación del anterior
  - No se permiten estructuras recursivas: p.ej.

$$\texttt{Nominal} \rightarrow \texttt{Nominal PP}$$

  - Ejemplos:

$$\begin{aligned} \texttt{NP} &\rightarrow \texttt{[Det] Noun* Noun} \\ \texttt{NP} &\rightarrow \texttt{Proper} \\ \texttt{VP} &\rightarrow \texttt{Verb} \\ \texttt{VP} &\rightarrow \texttt{Aux Verb} \end{aligned}$$

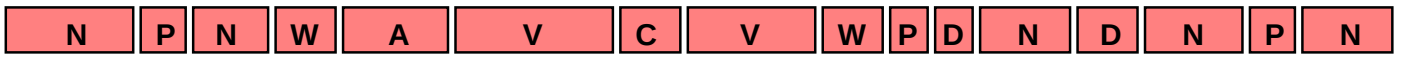
Mediante Correspondencia de Patrones (cont.)

- Pueden implementarse mediante **traductores finitos** (*finite-state transducers*, FSTs) emparentados con los autómatas finitos:
  - Eficiencia (complejidad lineal)
  - Simplicidad
- Pueden agruparse por niveles (i.e. en cascada) de forma que la salida del primer nivel sea la entrada al segundo, la salida del segundo la entrada al tercero... Esto permite:
  - Identificar estructuras cada vez más complejas
  - Generar estructuras arborescentes de altura limitada

Ejemplo (Vilares et al., 2008)

docenas de niños muy alegres han tenido que aprender hoy en el colegio una lección de historia

Etiquetador-Lematizador



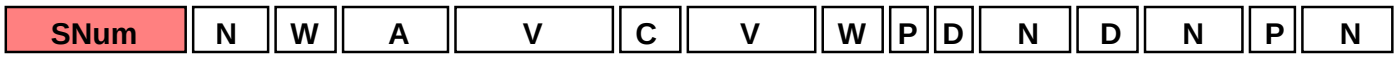
docena de niño muy alegre tener que aprender hoy en el colegio un lección de historia

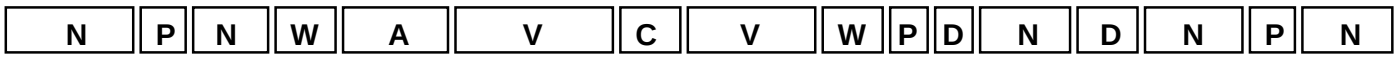
docenas de niños muy alegres han tenido que aprender hoy en el colegio una lección de historia

Nivel 0: Preprocesado

- Identificar expresiones numéricas y de cantidad (*NumP*)
  - e.g. `algo más de dos millones`
- Preprocesado de expresiones verbales: para simplificar el procesado en niveles superiores
  - e.g. `tener_en_cuenta` como unidad para evitar que `en cuenta` sea identificado como complemento del verbo

Nivel 0: Preprocesado





docena de niño muy alegre tener que aprender hoy en el colegio un lección de historia

docenas de niños muy alegres han tenido que aprender hoy en el colegio una lección de historia

Nivel 1: Frases Adverbiales y Verbos No-Perifrásticos

- Frases adverbiales ordinarias:

$$AdvP \rightarrow W^* \ W_1 \quad \begin{cases} AdvP.lem \doteq W_1.lem \\ AdvP.tag \doteq W_1.tag \end{cases}$$

- Grupos adjetivales con función adverbial, e.g. `de forma rápida` = `rápidamente`:

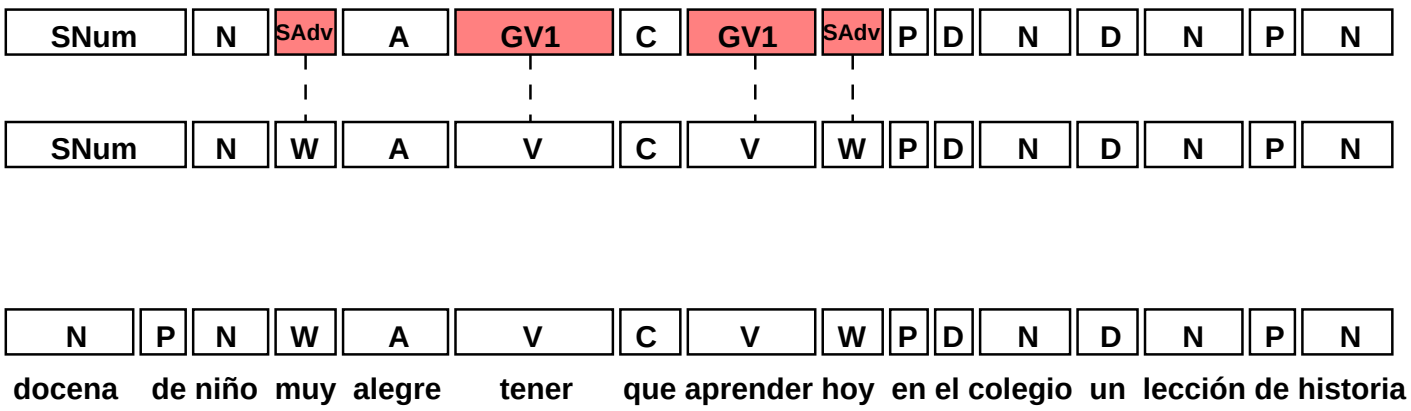
$$AdvP \rightarrow \texttt{de} \ (\texttt{forma} \mid \texttt{manera} \mid \texttt{modo}) \ A \quad \begin{cases} AdvP.lem \doteq A.lem \\ AdvP.tag \doteq A.tag \end{cases}$$

Nivel 1: Frases Adverbiales y Verbos No-Perifrásticos

- Formas activas y pasivas (verbo aux. `ser`)
- Tiempos simples y compuestos (verbo aux. `haber`)
- Ejemplo: formas compuestas pasivas

$$VG1 \rightarrow V_1 \; V_2 \; V_3 \begin{cases} VG1.lem \doteq V_3.lem \\ VG1.tag \doteq V_1.tag \\ VG1.voice \doteq \text{PASS} \\ V_1.lem \doteq \texttt{haber} \\ V_2.lem \doteq \texttt{ser} \\ V_2.tense \doteq \text{PART} \\ V_3.tense \doteq \text{PART} \end{cases}$$

Nivel 1: Frases Adverbiales y Verbos No-Perifrásticos



Nivel 2: Frases Adjetivales y Perífrasis Verbales

- Su núcleo es un adjetivo, que podría venir precedido por una frase adverbial:

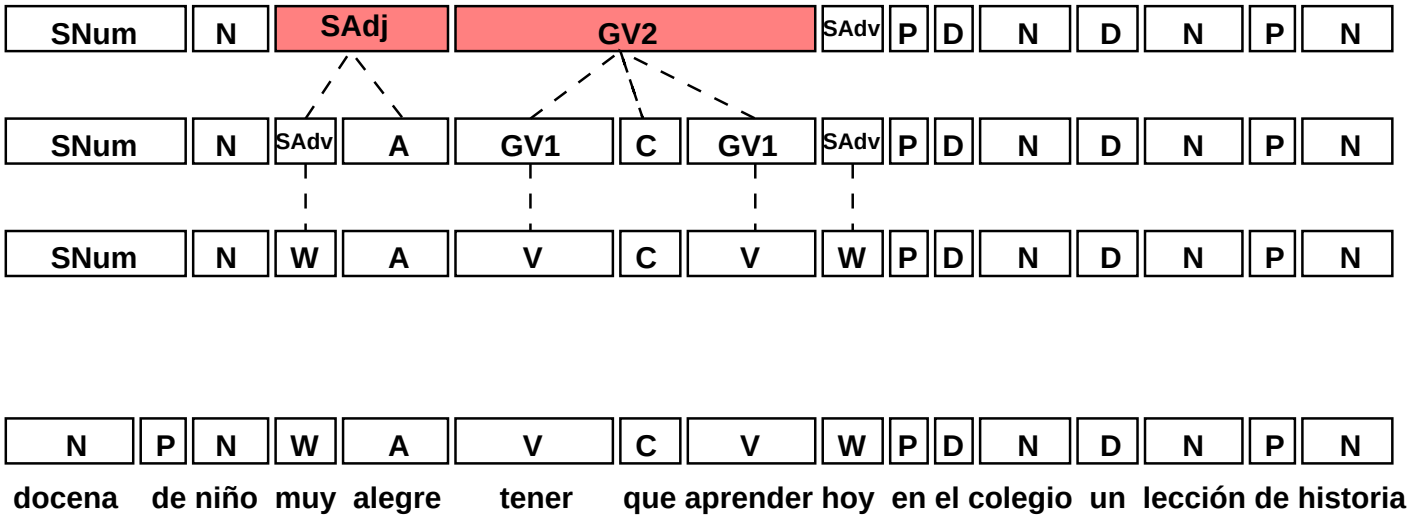
$$AdjP \rightarrow AdvP?\ A \begin{cases} AdjP.lem \doteq A.lem \\ AdjP.tag \doteq A.tag \end{cases}$$

Nivel 2: Frases Adjetivales y Perífrasis Verbales

- e.g. `tener+que+`*infinitivo*, `ir+a+`*infinitivo*
- Unión de dos o más formas verbales que funcionan como una unidad.
- Añaden matices de significado tales como obligación, grado de desarrollo de la acción, etc., que no pueden ser expresados mediante las formas verbales normales, simples o compuestas.
- Ejemplo: perífrasis de infinitivo

$$VG2 \rightarrow VG1_1 \ (\texttt{me}|\texttt{te}|\texttt{se})? \ (\texttt{que}|\texttt{de}|\texttt{a})? \ VG1_2$$
$$\begin{cases} VG2.lem \doteq VG1_2.lem \\ VG2.tag \doteq VG1_1.tag \\ VG2.voice \doteq VG1_2.voice \\ VG1_1.voice \doteq \text{ACT} \\ VG2_2.tense \doteq \text{INF} \end{cases}$$

Nivel 2: Frases Adjetivales y Perífrasis Verbales



Nivel 3: Frases Nominales

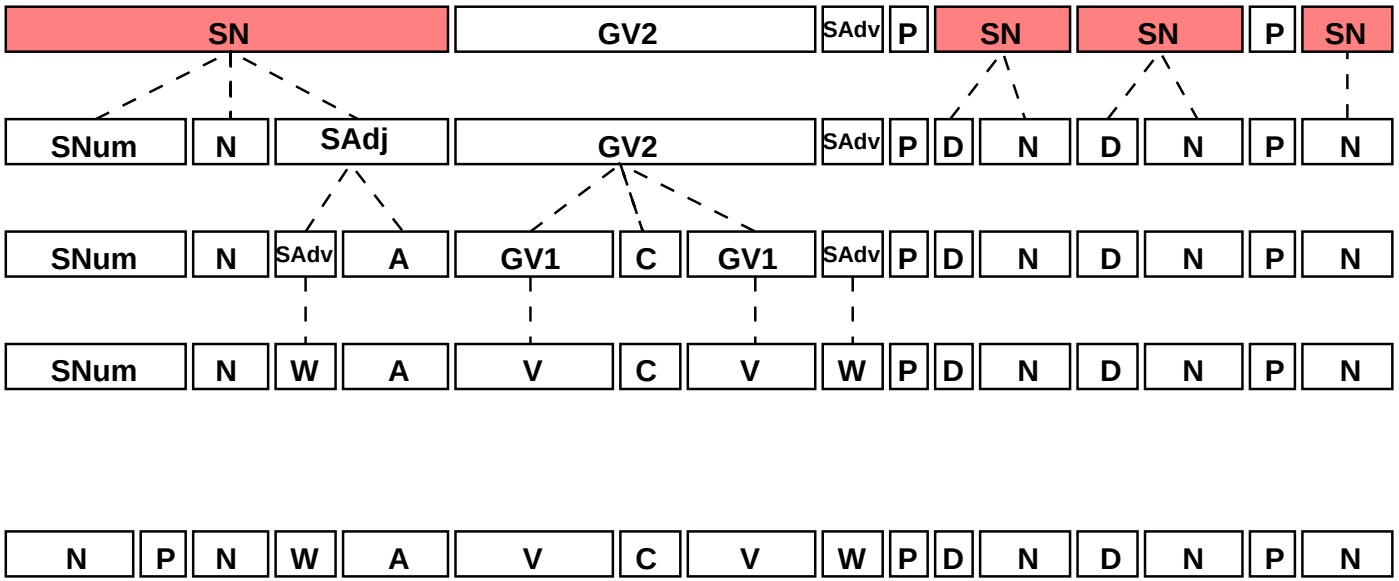
- Existencia de *complementos partitivos* (*PC*); e.g. `ninguno de`
- Secuencias/coordinaciones de frases adjetivales como post-modificadores (*AdjPostModif*)

$$\begin{aligned} AdjPostModif &\rightarrow AdjP\ \ Cc\ \ AdjP \\ &\mid\ AdjP \\ &\mid\ AdjP\ \ AdjP \\ &\mid\ AdjP\ \ AdjP\ \ AdjP \end{aligned}$$

- Existencia de posibles determinantes y frases adjetivales modificadoras antepuestos al núcleo nominal

$$\begin{aligned} NP \rightarrow\ & PC? \\ & D^*\ \ (AdjP \mid Number \mid NumP)? \\ & (N \mid Acronym \mid Proper)^* \\ & (N \mid Acronym \mid Proper)_1 \\ & AdjPostModif? \end{aligned} \quad \begin{cases} NP.lem \doteq ()_1.lem \\ NP.tag \doteq ()_1.tag \\ NP.num \doteq PC.num \end{cases}$$

Nivel 3: Frases Nominales



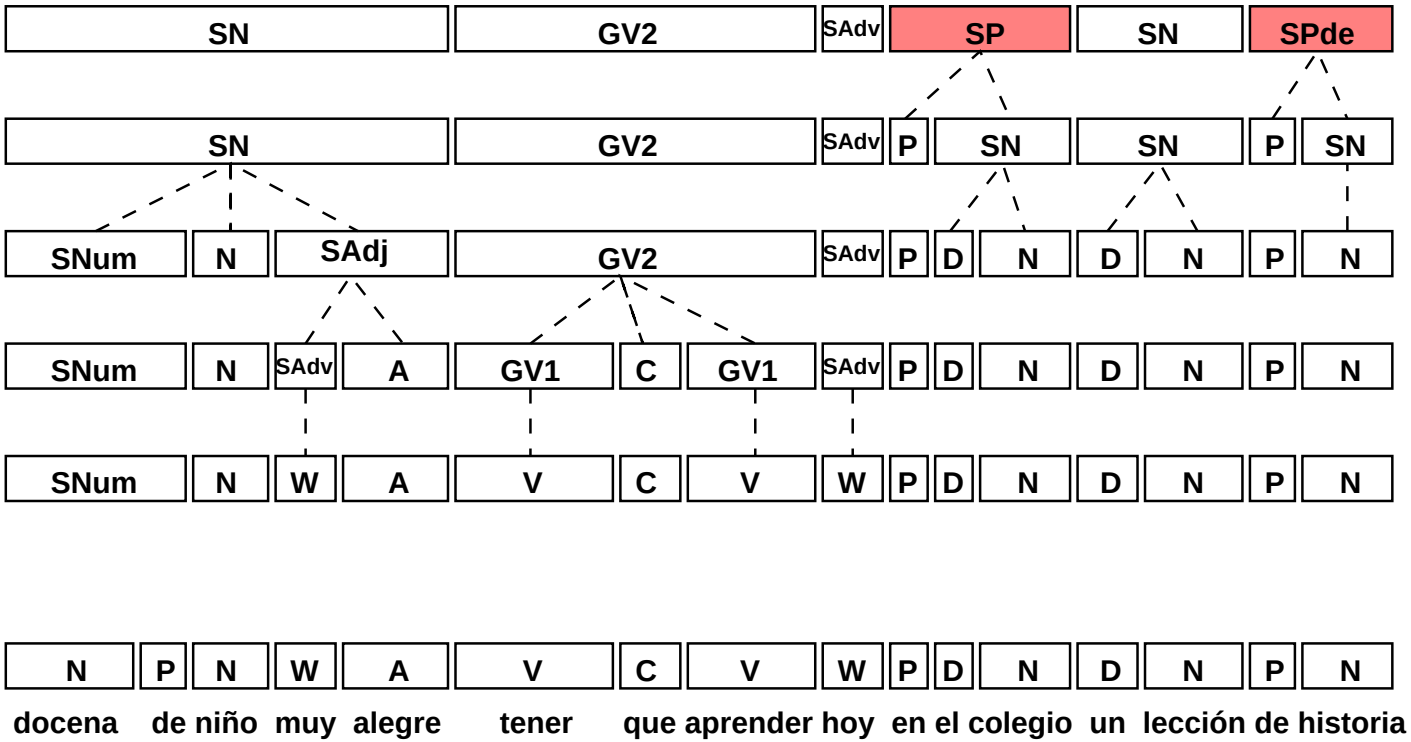
docenas de niños muy alegres han tenido que aprender hoy en el colegio una lección de historia

Nivel 4: Frases Preposicionales

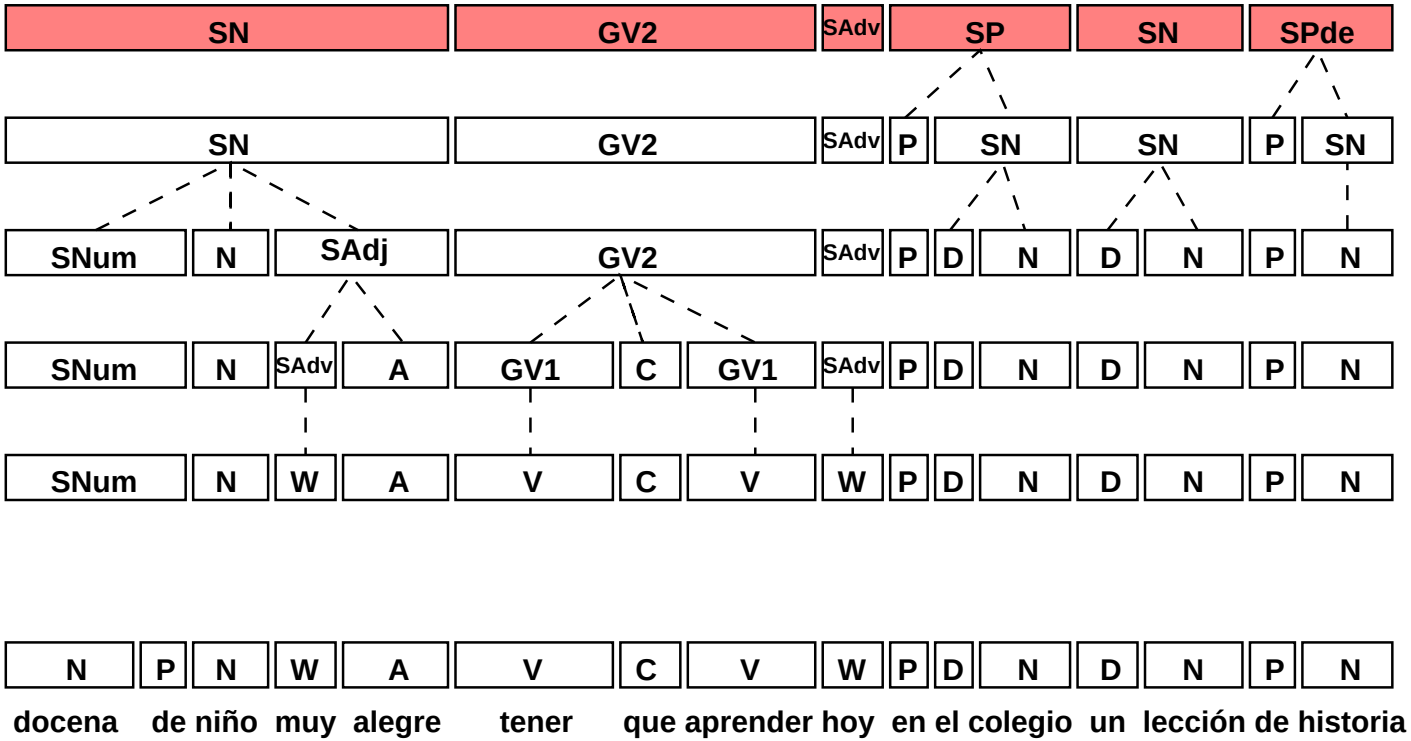
- Para facilitar la extracción de términos en fases posteriores distinguiremos 3 tipos según la preposición:
  - *PPde*: preposición `de`
  - *PPpor*: preposición `por`
  - *PP*: otras
- Ejemplo: frases preposicionales introducidas mediante `de`

$$PPof \rightarrow P \;\; NP \begin{cases} P.lem \doteq \texttt{de} \\ PP.lem \doteq NP.lem \\ PP.tag \doteq NP.tag \end{cases}$$

Nivel 4: Frases Preposicionales



Resultado final del análisis



Mediante Reglas Aprendidas Automáticamente

- Similar al *etiquetador de Brill* pero para IOB tagging:
  - Etiqueta inicial: en base a la etiqueta morfosintáctica (*part-of-speech/PoS tag*) de la palabra
    - Se le asigna el *chunk tag* (I,O,B) más frecuente para esa categoría
  - Reglas de transformación: en base a la forma, etiqueta morfosintáctica y *chunk tag* actuales de la palabra y sus contiguas

Mediante Reglas Aprendidas Automáticamente (cont.)

- Necesidad de un **corpus de entrenamiento**:
  - Textos con las frases de interés previamente delimitadas y etiquetadas
  - Problema: muy costoso de crear
  - Solución: reutilizar *treebanks* ya existentes
    - Se toma un árbol sintáctico del *treebank*
    - Se identifican sus frases/grupos básicos (NP, VP, PP, ...) no recursivos
    - Se [re]anotan convenientemente

Ejemplo de Reglas Aprendidas Automáticamente

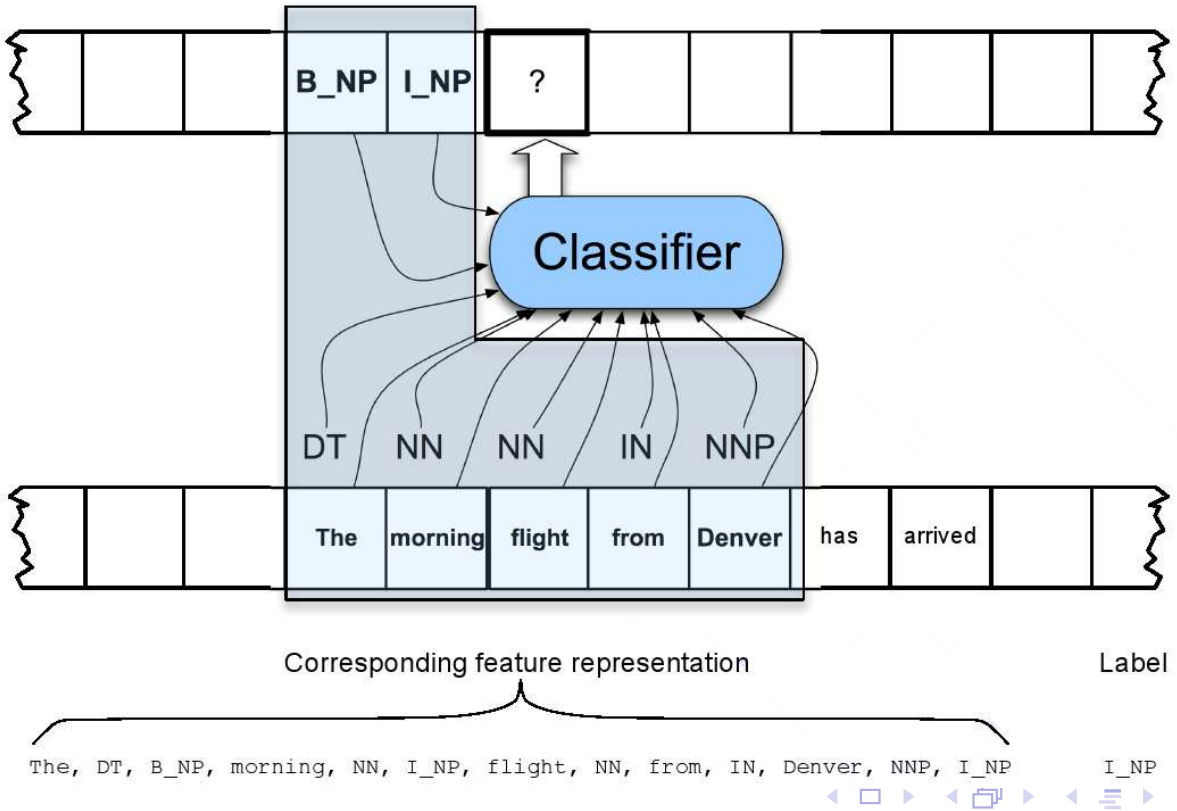
W_0, W_{-1}, W_1:	palabras actual, a la izquierda y a la derecha, respectivamente
P_0, P_{-1}, P_1:	ídem para las etiquetas morfosintácticas
T_0, T_{-1}, T_1:	ídem para las *chunk tags*

Pasada	Anterior	Contexto	Nueva
1	I	T_1=O, P_0=ADJ	O
2	-	T_{-2}=I, P_{-1}=I, P_0=DET	B
		...	

1. Una *chunk tag* I pasa a ser O cuando la etiqueta morfosintáctica de la palabra actual es un adjetivo (ADJ) y la siguiente palabra tiene un *chunk tag* O.
2. Asignamos una *chunk tag* B a la palabra actual si los *chunk tag* de las dos palabras anteriores son I y la etiqueta morfosintáctica de la palabra actual es un determinante (DET)

Mediante Clasificadores Secuenciales

- P.ej. etiquetación estocástica basada en modelos de Markov
- Aproximaciones posibles (ya introducidas):
  - Etiquetación de palabras
  - Etiquetación de separaciones entre palabras (i.e. parentización)
- **Etiquetación de palabras**: etiquetar cada palabra en base a la forma/lema/stem, etiqueta morfosintáctica y *chunk tag* de ella misma y sus contiguas



Mediante Clasificadores Secuenciales (cont.)

- **Etiquetación de separaciones**: determinar la secuencia de *gap tags* $G = g_2, g_3 \ldots g_n$ óptima en función de las etiquetas morfosintácticas $T = t_1, t_2 \ldots t_n$ y las formas $W = w_1, w_2 \ldots w_n$ de las palabras que separan:
  - i.e. maximizar $P(G) = \prod_{i=2}^{n} P(g_i | w_{i-1}, t_{i-1}, w_i, t_i)$

Demos on-line

- Freeling 2.1 (incluyendo español y gallego):
  `http://garraf.epsevg.upc.es/freeling/demo.php`
- Cognitive Computation Group (CCG), Univ. of Illinois at Urbana-Champaign:
  `http://l2r.cs.uiuc.edu/~cogcomp/shallow_parse_demo.php`
- Memory-Based Shallow Parsing (MBSP) demo, Computational Linguistics & Psycholinguistics (CLiPS) Research Centre, University of Antwerp:
  `http://www.cnts.ua.ac.be/cgi-bin/jmeyhi/MBSP-instant-webdemo.cgi`

Referencias

- [Abney, 1997] Abney, S. (1997). Partial Parsing via Finite-State Cascades. In *Natural Language Engineering*, 2(4), 337–244.
- [Jurafsky & Martin, 2009] Jurafsky, D. & Martin, J.H. (2009). Chapter 13: Syntactic Parsing. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (2nd ed.)*. Pearson–Prentice Hall.
- [Nugues, 2006] Nugues, P.M. (2006). Chapter 9: Partial Parsing. *An Introduction to Language Processing with Perl and Prolog*. Springer-Verlag.
- [Vilares et al., 2008] Vilares, J., Alonso, M.A. & Vilares, M. (2008). Extraction of Complex Index Terms in Non-English IR: A Shallow Parsing Based Approach. *Information Processing & Management*, 44(4), 1517–1537.