

IMPLEMENTACIÓN DE UNA GRAMÁTICA DE CLÁUSULAS DEFINIDAS EN PROLOG

PL2 – SWI Prolog – 5 de abril de 2017

Conocimiento Y Razonamiento Automatizado

Miguel Romeral

Contenido

Objetivo	2
Gramática	2
Criterios de diseño de la gramática	5
Ejemplos de ejecución.....	6
Del punto 3:.....	6
Del punto 7:.....	9
Otros ejemplos:	14

Objetivo

La aplicación en Prolog simula el análisis sintáctico de diferentes oraciones cuyas palabras se encuentren en el diccionario de la aplicación. Es importante destacar que la aplicación indica qué tipo de sintagma o palabra es, sin embargo, no indica cuál es su funcionalidad, tales como sujeto, complemento directo, complemento circunstancial de modo, etc.

Gramática

Partiendo de la gramática que se indicaba en el enunciado de la práctica, se han desarrollado diferentes reglas que completan las diferentes reglas de la gramática castellana. Debido a la enorme dificultad de simular una gramática completa y la destacada ambigüedad del lenguaje, el analizador de las cláusulas definidas obtiene una única interpretación de la frase que se intente analizar por causa de la arquitectura de Prolog.

La leyenda de la gramática que se explica seguidamente es la siguiente:

LEYENDA:

NO TERMINALES:	
<i>O</i>	Oración
<i>GN</i>	Sintagma Nominal.
<i>GV</i>	Sintagma Verbal.
<i>OR_SUB_AUX</i>	Oración cuyo sujeto tiene una subordinada.
<i>OC</i>	Oración compuesta.
<i>GN_S</i>	Sintagma Nominal con oración subordinada.
<i>OS</i>	Oración Subordinada.
<i>GN_C</i>	Continuación del GN (sujeto múltiple).
<i>GN_ADJ</i>	Sintagma Adjettival.
<i>GP_AUX</i>	Sintagma Preposicional (uno o más).
<i>VC</i>	Verbo (simple o compuesto)
<i>SA</i>	Sintagmas (uno o más)
<i>S</i>	Sintagma.
<i>GADV</i>	Sintagma Adverbial.
<i>GP</i>	Sintagma Preposicional.
TERMINALES:	
<i>conj</i>	Conjunción
<i>det</i>	Determinante.
<i>n</i>	Nombre.
<i>prep</i>	Preposición
<i>c</i>	Cuantificador
<i>adv</i>	Adverbio
<i>adj</i>	Adjetivo
<i>v</i>	Verbo

A continuación, se muestra resumidas las reglas gramaticales de las que está compuesto el programa. (Las palabras en mayúsculas indican símbolos no terminales y en minúsculas los terminales).

O → GN, GV

O → GV

$O \rightarrow OR_SUB_AUX$

$O \rightarrow OR_SUB_AUX, conj, O$

$O \rightarrow OC$

Podemos observar que el axioma, el símbolo por el que se empieza a derivar, es O.

La cuarta regla que acabamos de ver corresponde a la creación de oraciones compuestas, las cuales tienen una oración subordinada y otra oración separadas por una conjunción.

$OR_SUB_AUX \rightarrow GN_S, GV$

$GN_S \rightarrow GN, OS$

Algunas oraciones tienen la oración subordinada en el sujeto, por lo que se encuentre en el sintagma nominal. Se hace la distinción mediante símbolos auxiliares.

$OS \rightarrow conj, GV$

La oración subordinada está compuesta por la conjunción “que” seguido de un sintagma verbal. El motivo por el que no es una conjunción seguido de una oración es que, si la oración es compuesta, aparecerían errores de desplazamiento-reducción al no saber qué derivar en caso de que después existiera otra oración seguido por una conjunción, puesto que el ordenador no sabría si esa oración pertenece a la subordinada o a la oración principal.

$GN \rightarrow n$

$GN \rightarrow n, GN_C$

$GN \rightarrow det, n$

$GN \rightarrow det, n, GN_C$

$GN \rightarrow n, GADJ$

$GN \rightarrow det, n, GP_AUX$

$GN \rightarrow det, n, GADJ$

$GN \rightarrow det, GADJ, n$

Aquí se aprecian los diferentes tipos de sintagmas nominales que existen en la gramática.

$GN_C \rightarrow conj GN$

$GN_C \rightarrow conj, GN, GN_C$

Estas reglas auxiliares sirven para indicar varios sujetos, ejemplo: “Juan, Héctor y María”, aunque como nuestro programa no detecta comas (ya que es nuestro separador de palabras) se indica con “y”, quedando, por tanto: “Juan y Héctor y María”.

$GV \rightarrow VC$

GV → VC, SA

Los sintagmas verbales tienen un verbo (VC se encarga de determinar si es compuesto o no) y cero o más sintagmas de diferente índole.

SA → S

SA → S, SA

De esta manera se indica la continuación indefinida de sintagmas en el verbal. Se aprecia claramente la eliminación de la recursividad por la izquierda, la cual potencia los bucles en Prolog.

S → GN

S → GADJ

S → GADV

S → GP

S → OS

Aquí se muestran los cinco tipos de sintagmas que detecta el analizador sintáctico.

GP_AUX → GP

GP_AUX → GP, GP_AUX

Continuación del sintagma preposicional, pudiendo poner de uno a nueve sintagmas preposicionales, ejemplo: “en el jardín del vecino de Barcelona” tiene tres sintagmas preposicionales, los cuales cuentan como uno, ya que “el vecino de Barcelona” no tendría sentido si no hablásemos de su jardín.

OC → O conj O

OC → O conj OC

Así se asegura la continuación de diferentes oraciones coordiadas, todas ellas separadas por conjunciones “y”.

GP → prep, GN

GP → prep, GV

El sintagma preposicional puede hacer referencia a uno nominal o verbal.

GADV → adv

GADV → c, adv

GADJ → adj

GADJ \rightarrow c, adj

Los sintagmas adjetivales y adverbiales tienen la misma forma, pudiendo tener o no un cuantificador.

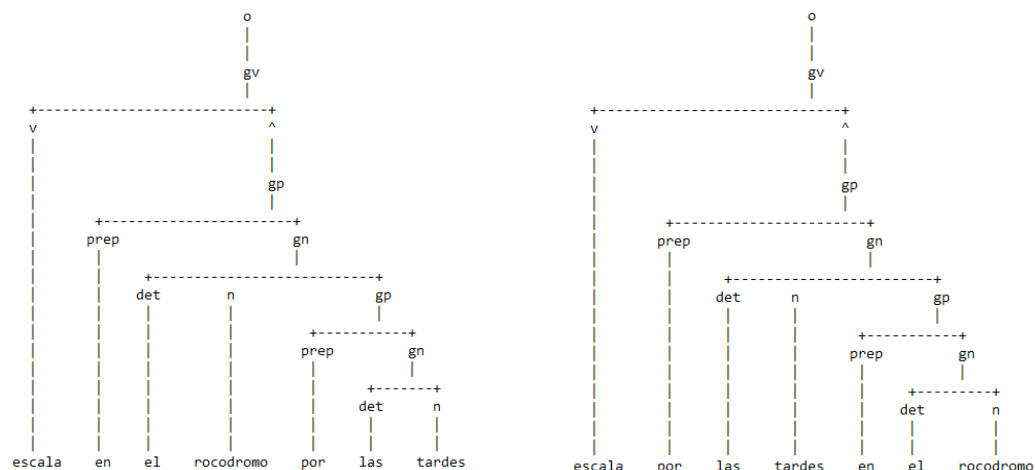
VC \rightarrow v

VC \rightarrow v, v

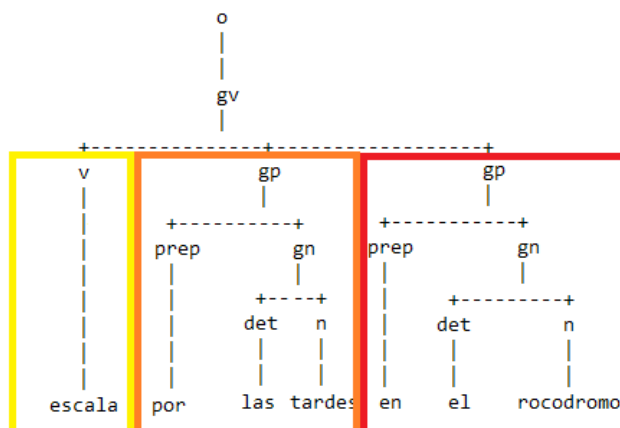
Se determina si un verbo es compuesto o simple.

Criterios de diseño de la gramática

- Hay que mencionar la necesidad de evitar la recursividad por la izquierda en estas reglas, puesto que produce bucles potenciales en diferentes entradas en el lenguaje Prolog, como ya se mencionó anteriormente.
- Algunos sintagmas preposicionales, como los demás sintagmas, que se encuentran en sintagmas verbales no están al mismo nivel que el núcleo del sintagma (en este caso el verbo), la aplicación tiene en cuenta la continuación de sintagmas que no se encuentran en el mismo nivel. Un ejemplo es “escala en el rocódromo por las tardes”, se podría separar por “escala por las tardes en el rocódromo”, sin embargo, los árboles que se generan son:

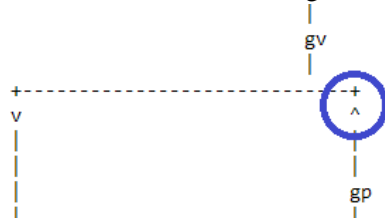


donde se puede apreciar que el segundo sintagma preposicional se analiza como si estuviera referenciado por el primero, cuando en realidad debería quedar de la siguiente forma:



donde los dos sintagmas se encuentran a la misma altura e independientes entre ellos, puesto que el naranja es un complemento circunstancial de tiempo y el rojo de lugar.

- Tras varias pruebas, la recursividad para poder indicar ene sintagmas en un sintagma verbal (no un número predefinido), no se consiguió obtener una representación del árbol como la de la anterior figura. Sin embargo, se indica mediante el siguiente símbolo:



indicando que ese sintagma debe estar en el mismo nivel que el núcleo. Si existen más sintagmas se indican de igual manera.

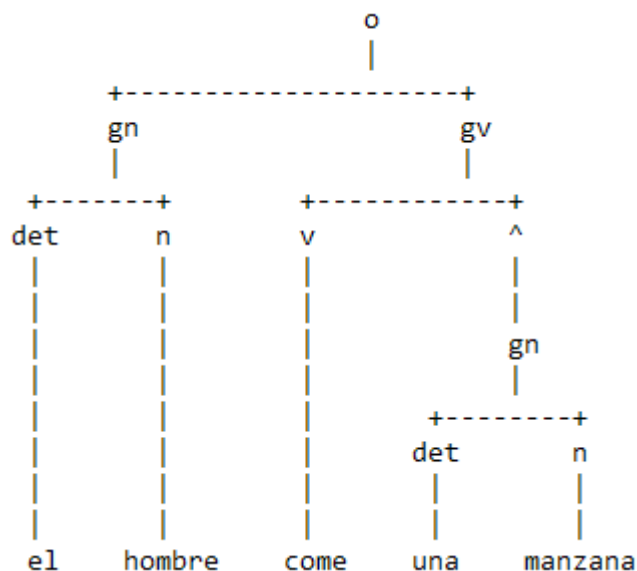
- En las oraciones subordinadas no se permiten oraciones coordinadas, puesto que, como se indicó en la gramática, generan conflictos sobre qué reglas derivar, ya que las conjunciones pueden referirse a la oración principal o a la subordinada. Por ese motivo, sentencias más ambiguas han sido desestimadas.

Ejemplos de ejecución.

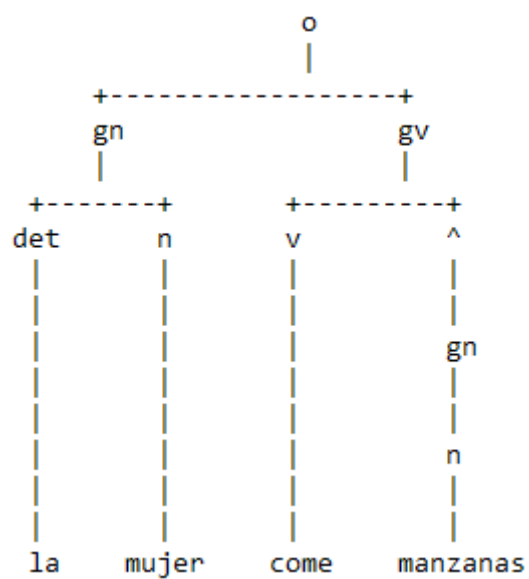
Además de los ejemplos de ejecución que venían a lo largo del enunciado de la práctica, varias han sido las oraciones que, tras haber sido añadidas al diccionario todas sus palabras, se han probado, el código a ejecutar desde la línea de comandos, tras cargar ambos archivos ([analizador],[draw].) es:

Ejemplos del punto 3:

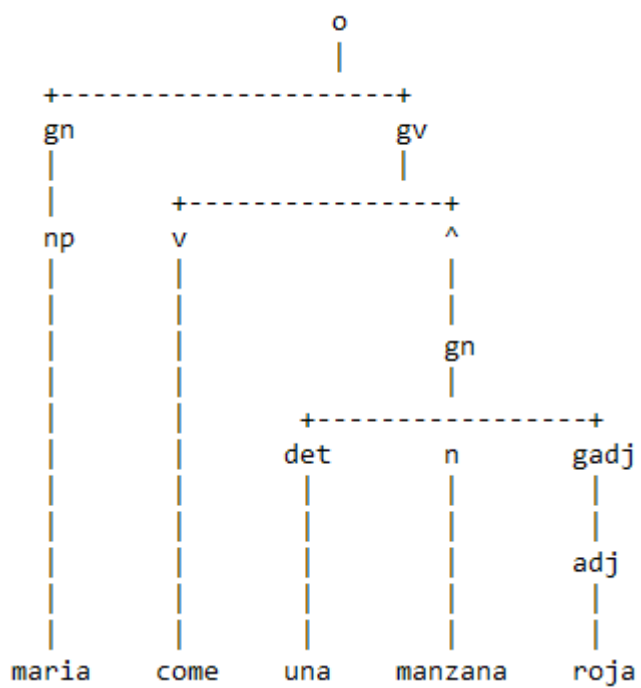
oracion(X,[el,hombre,come,una,manzana],[]),draw(X).



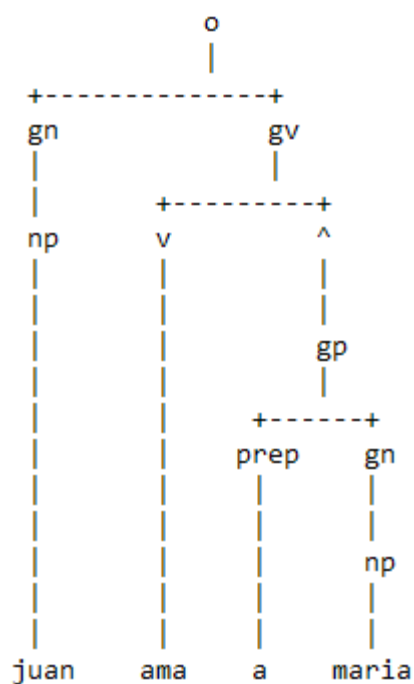
oracion(X,[la,mujer,come,manzanas],[]),draw(X).



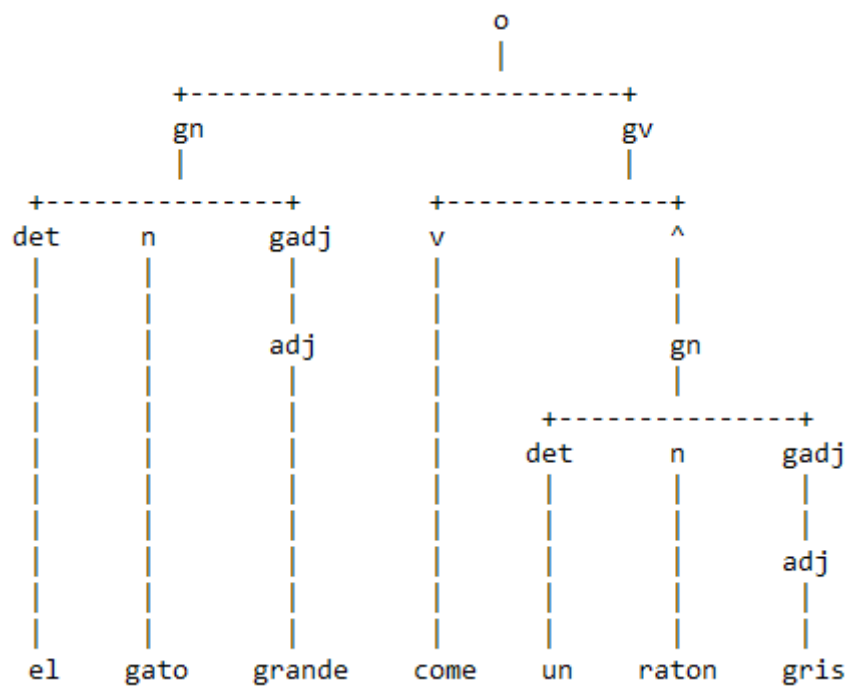
oracion(X,[maria,come,una,manzana,roja],[]),draw(X).



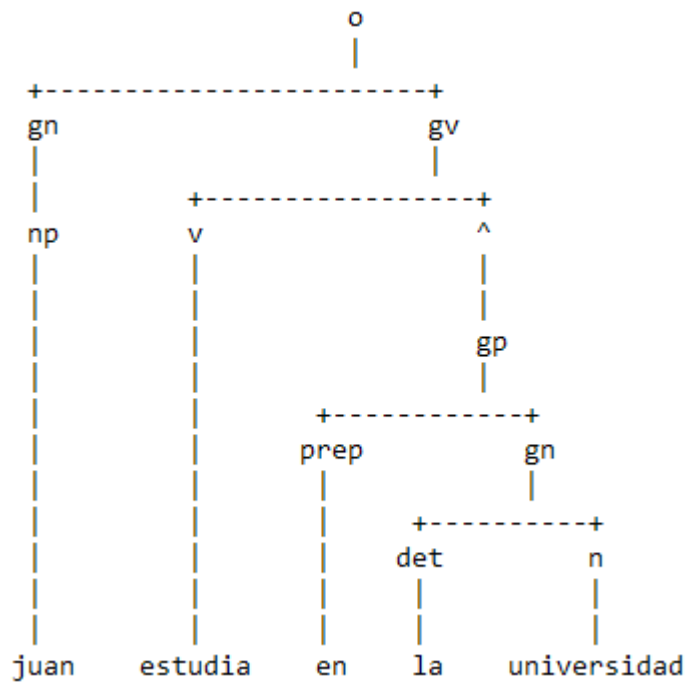
oracion(X,[juan,ama,a,maria],[]),draw(X).



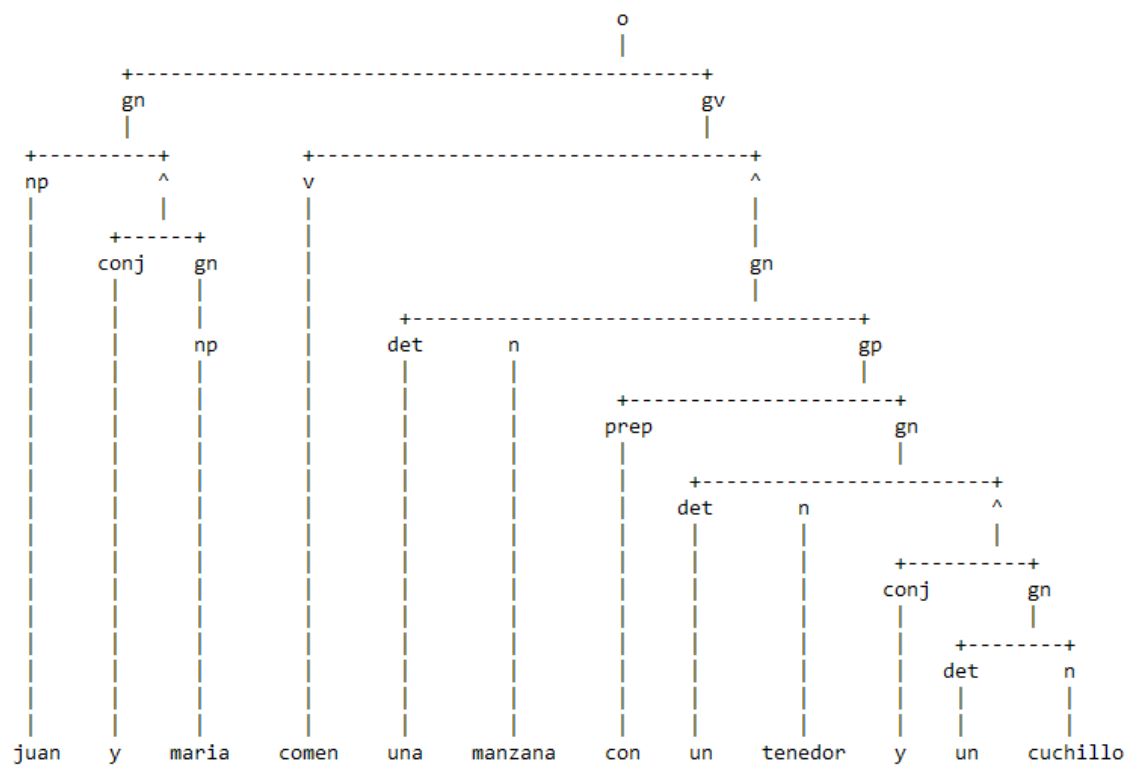
oracion(X,[el,gato,grande,come,un,raton,gris],[]),draw(X).



oracion(X,[juan,estudia,en,la,universidad],[]),draw(X).

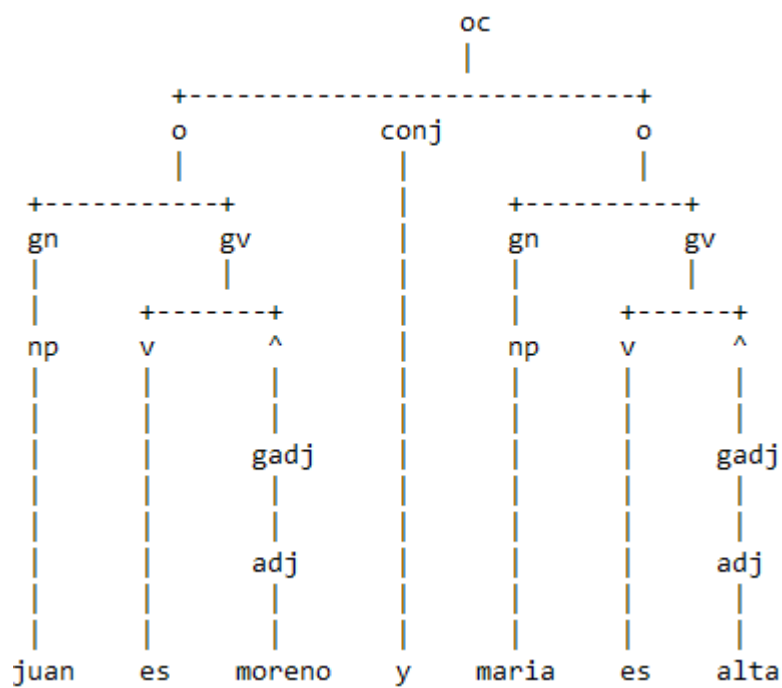


oracion(X,[juan,y,maria,comen,una,manzana,con,un,tenedor,y,un,cuchillo],[]),draw(X).

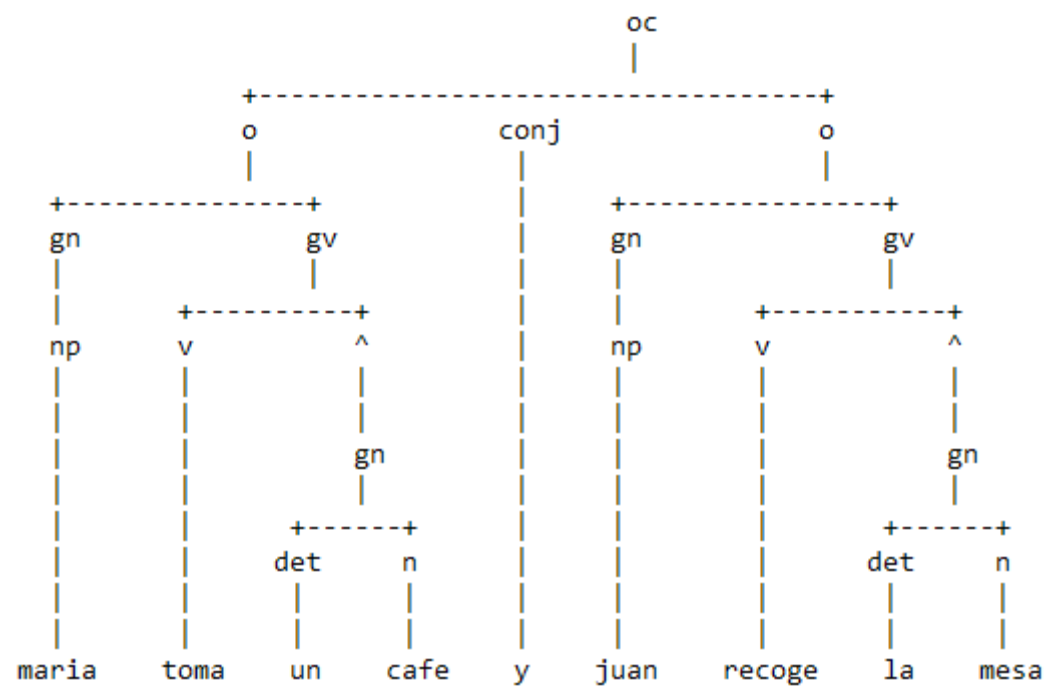


Del punto 7:

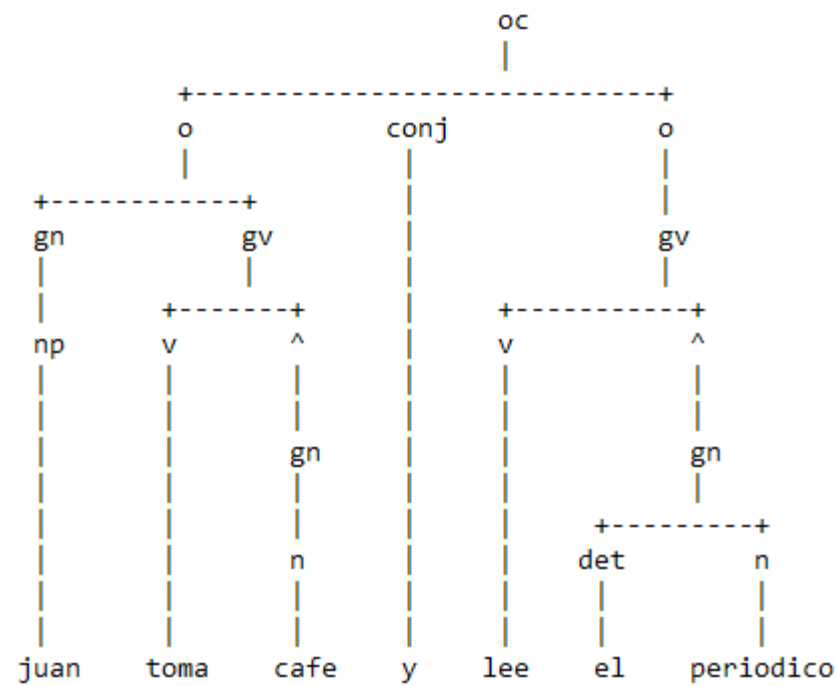
oracion(X,[juan,es,moreno,y,maria,es,alta],[]),draw(X).



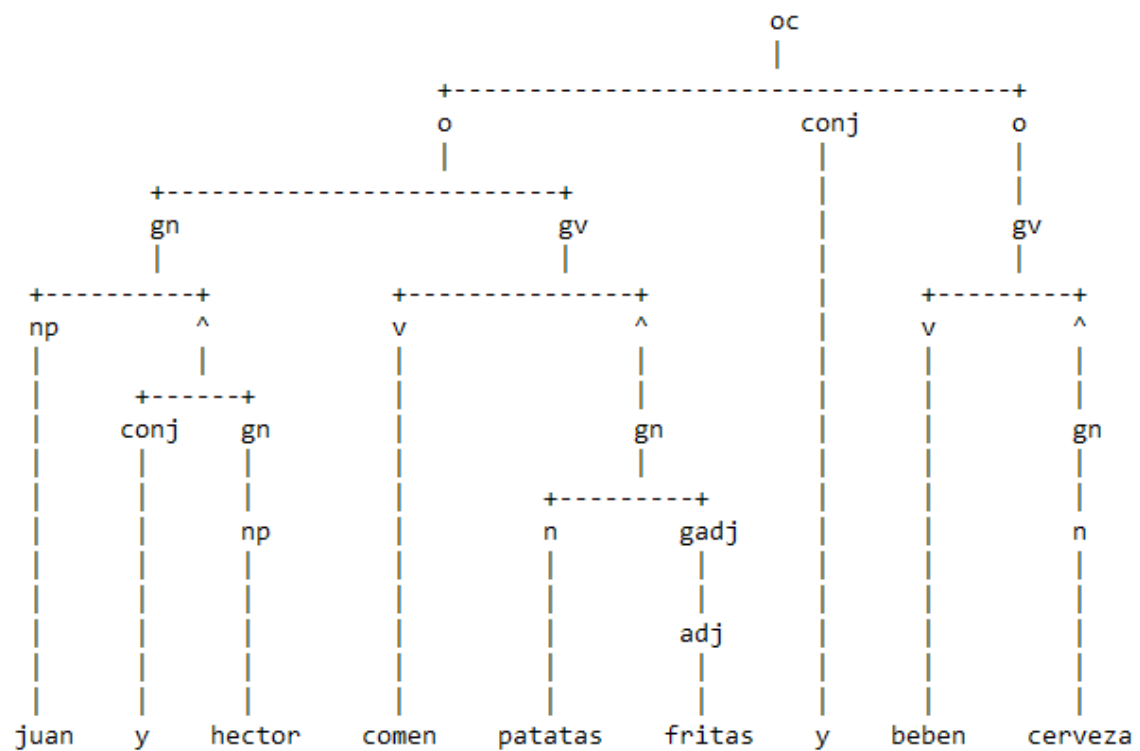
oracion(X,[maria,toma,un,cafe,y,juan,recoge,la,mesa],[]),draw(X).



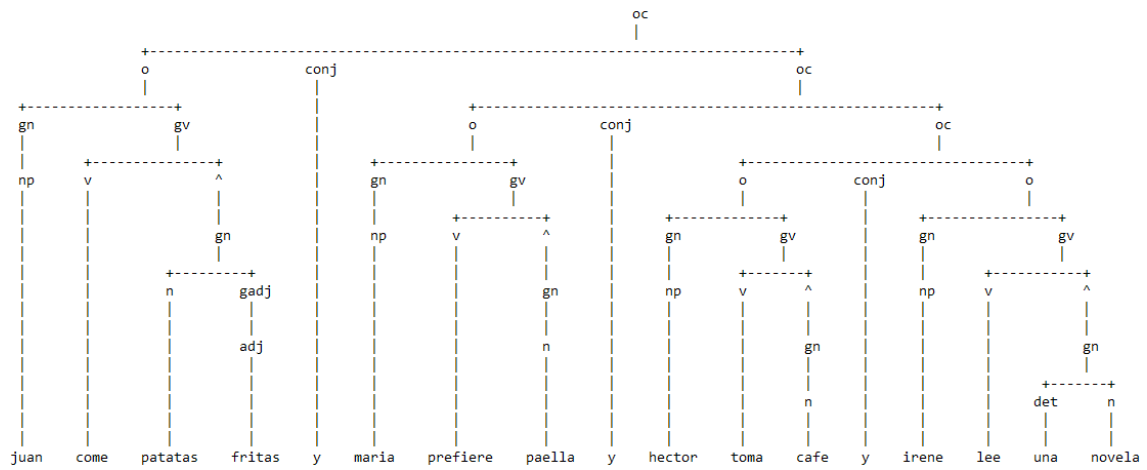
oracion(X,[juan,toma,cafe,y,lee,el,periodico],[]),draw(X).



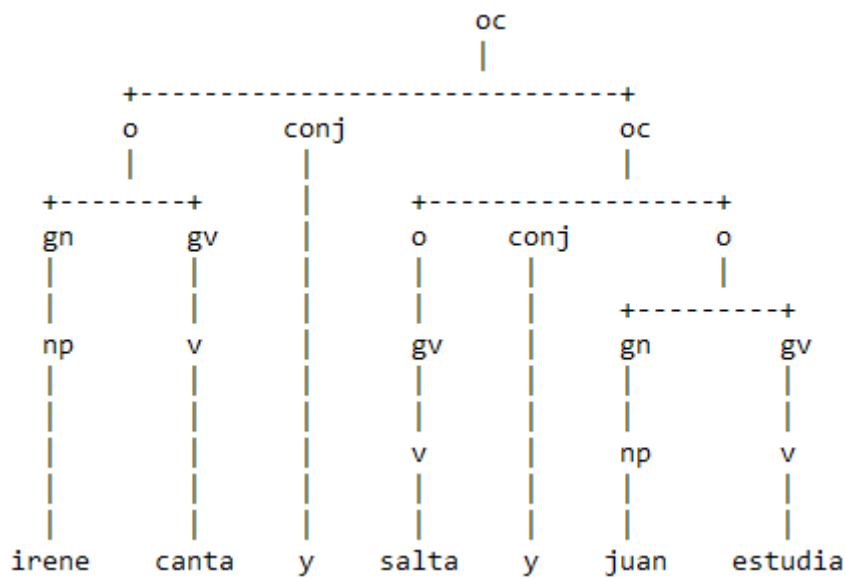
oracion(X,[juan,y,hector,comen,patatas,fritas,y,beben,cerveza],[]),draw(X).



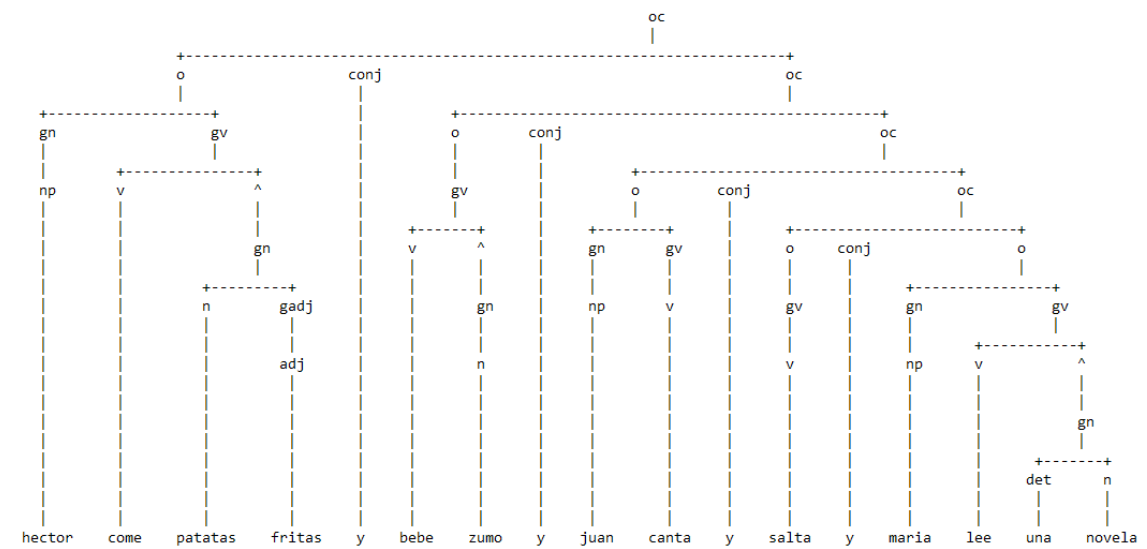
oracion(X,[juan,come,patatas,fritas,y,maria,prefiere,paella,y,hector,toma,cafe,y,irene,lee,una,novela],[]),draw(X).



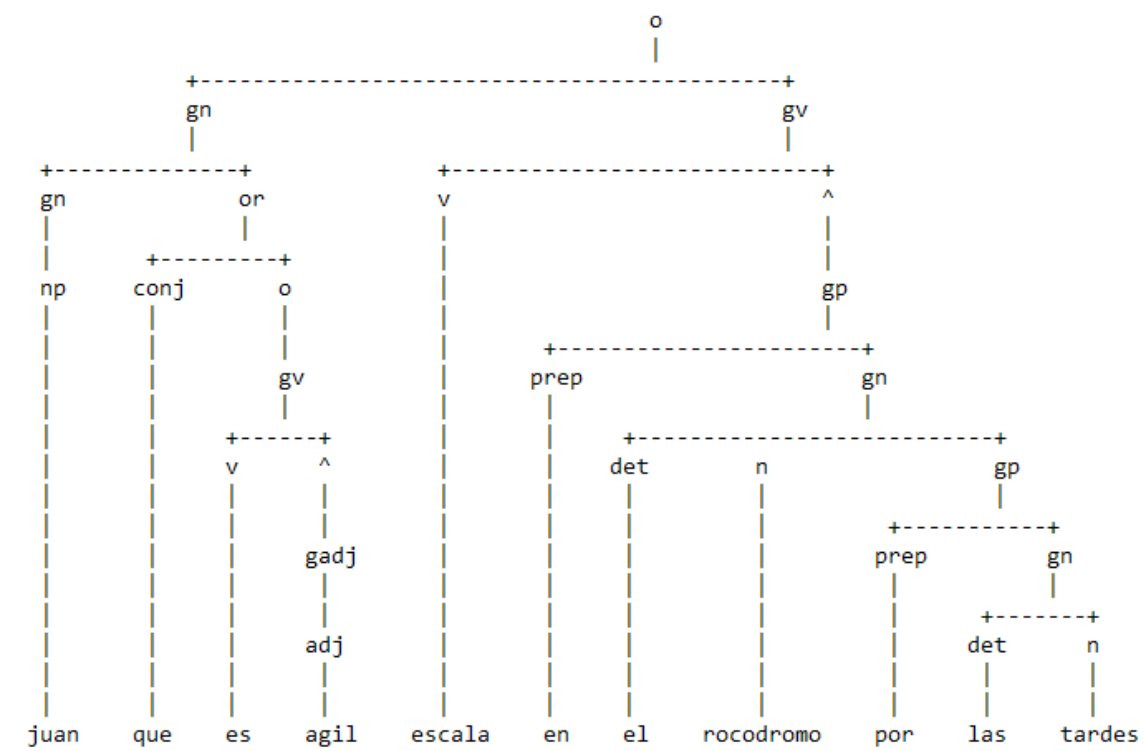
oracion(X,[irene,canta,y,salta,y,juan,estudia],[]),draw(X).



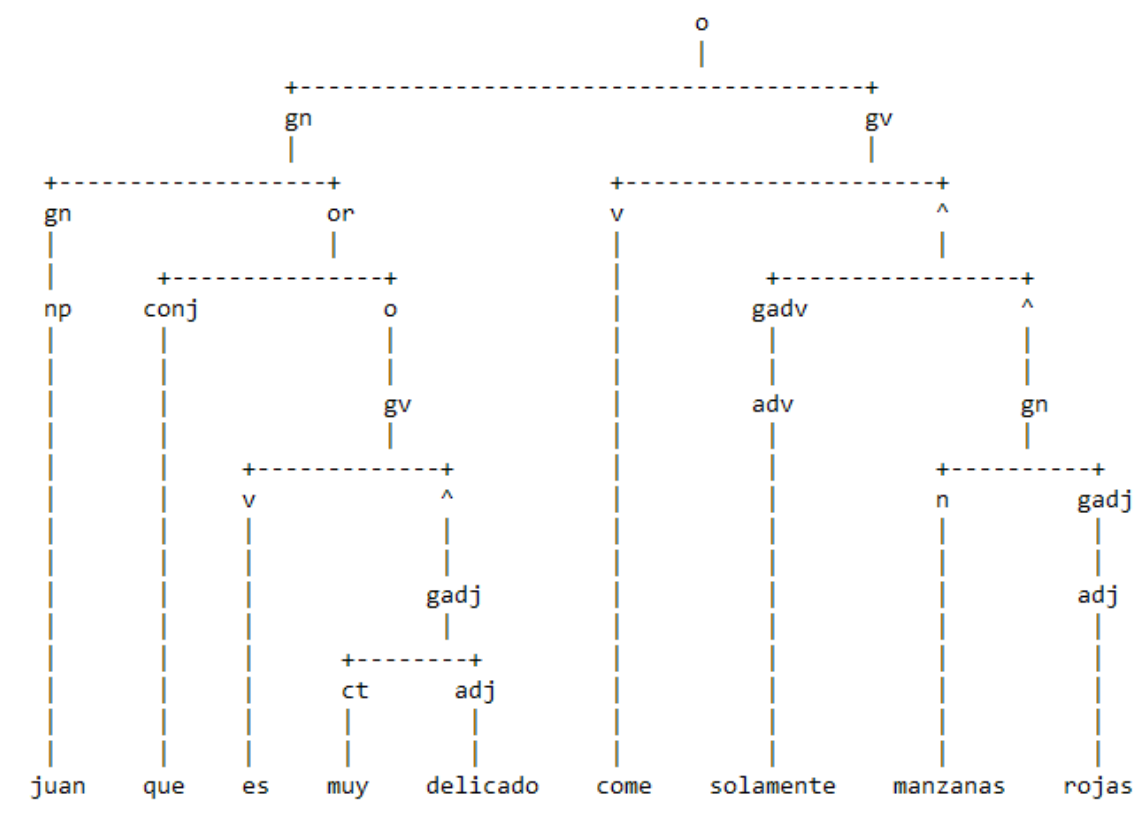
oracion(X,[hector,come,patatas,fritas,y,bebe,zumo,y,juan,canta,y,salta,y,maria,lee,una,novela],[]),draw(X).



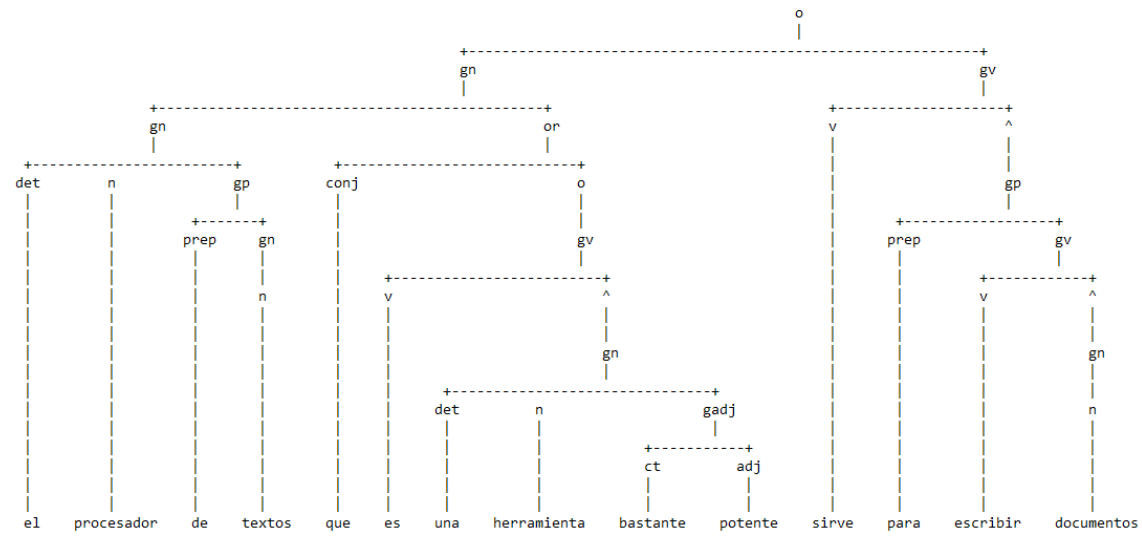
oracion(X,[juan,que,es,agil,escala,en,el,rocodromo,por,las,tardes],[
 ,draw(X).



oracion(X,[juan,que,es,muy,delicado,come,solamente,manzanas,rojas],[
 ,draw(X).

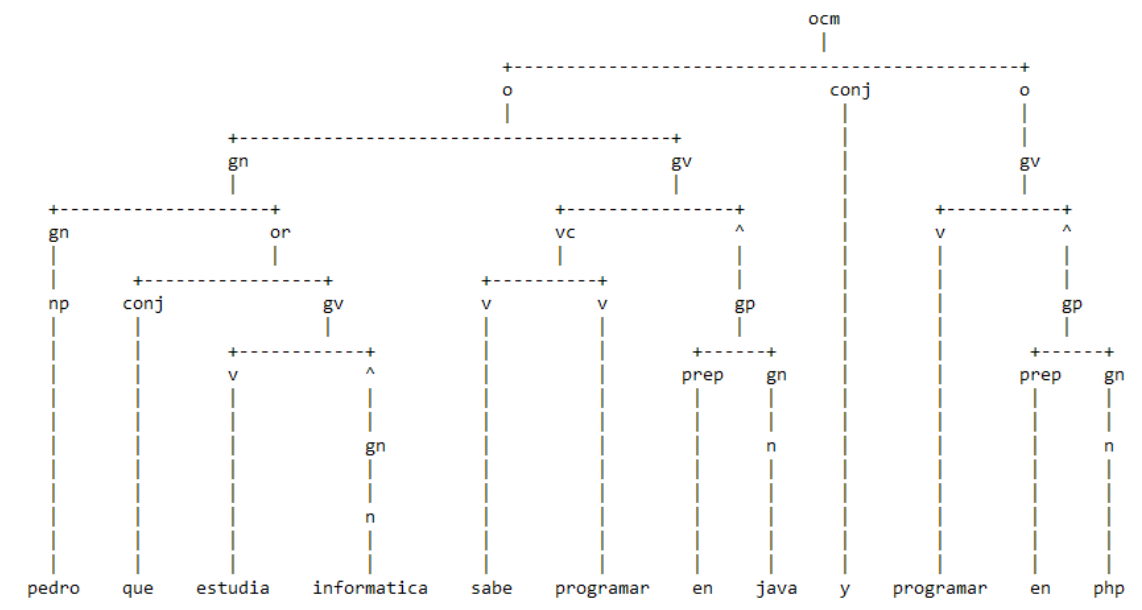


```
oracion(X,[el,procesador,de,textos,que,es,una,herramienta,bastante,pot
ente,sirve,para,escribir,documentos],[]),draw(X).
```

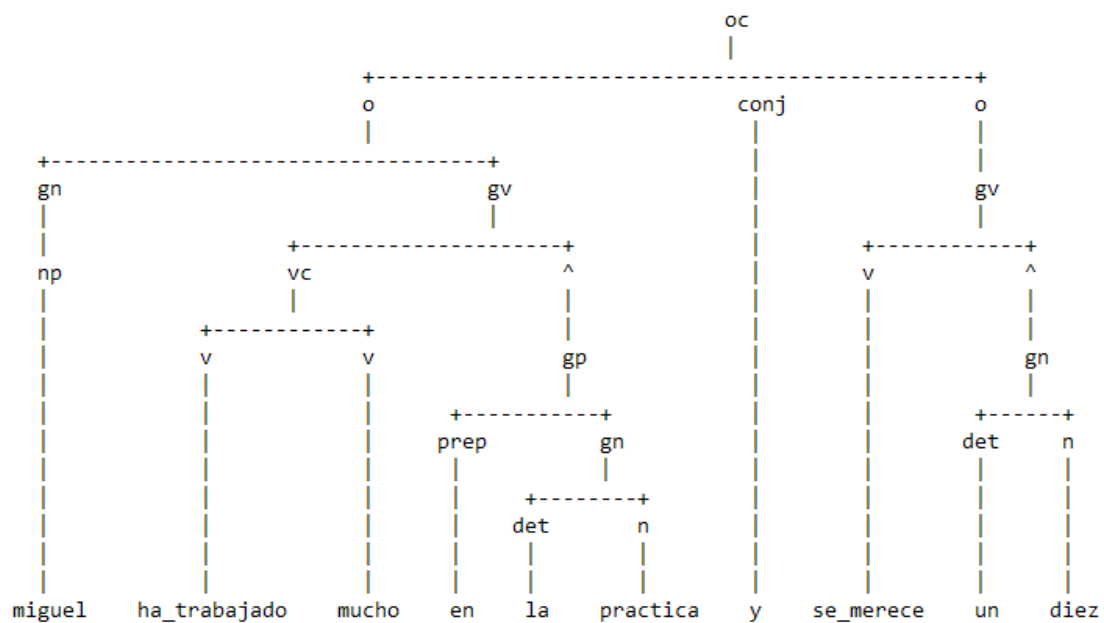


Otros ejemplos:

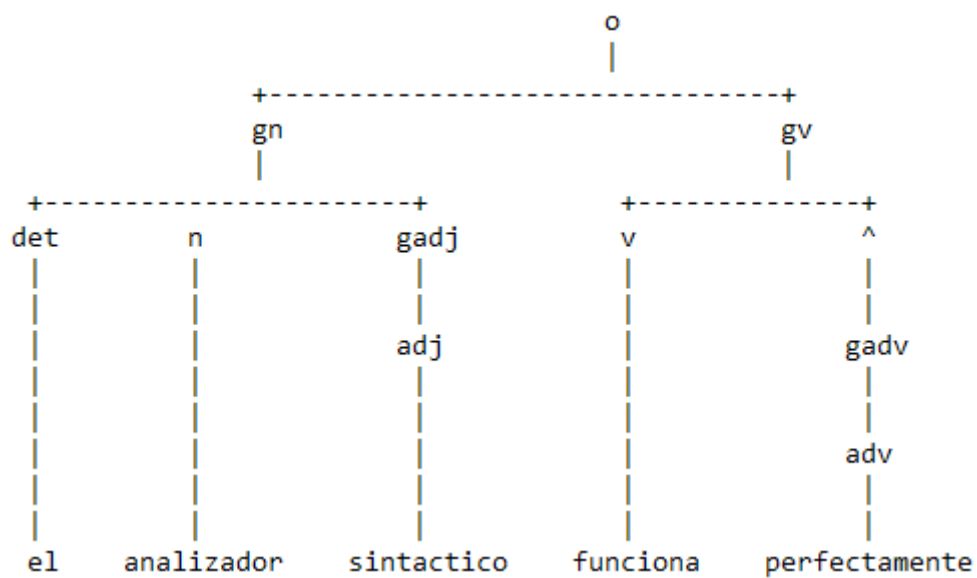
```
oracion(X,[pedro,que,estudia,informatica,sabe,programar,en,java,y,programar,en,php],[ ]),draw(X).
```



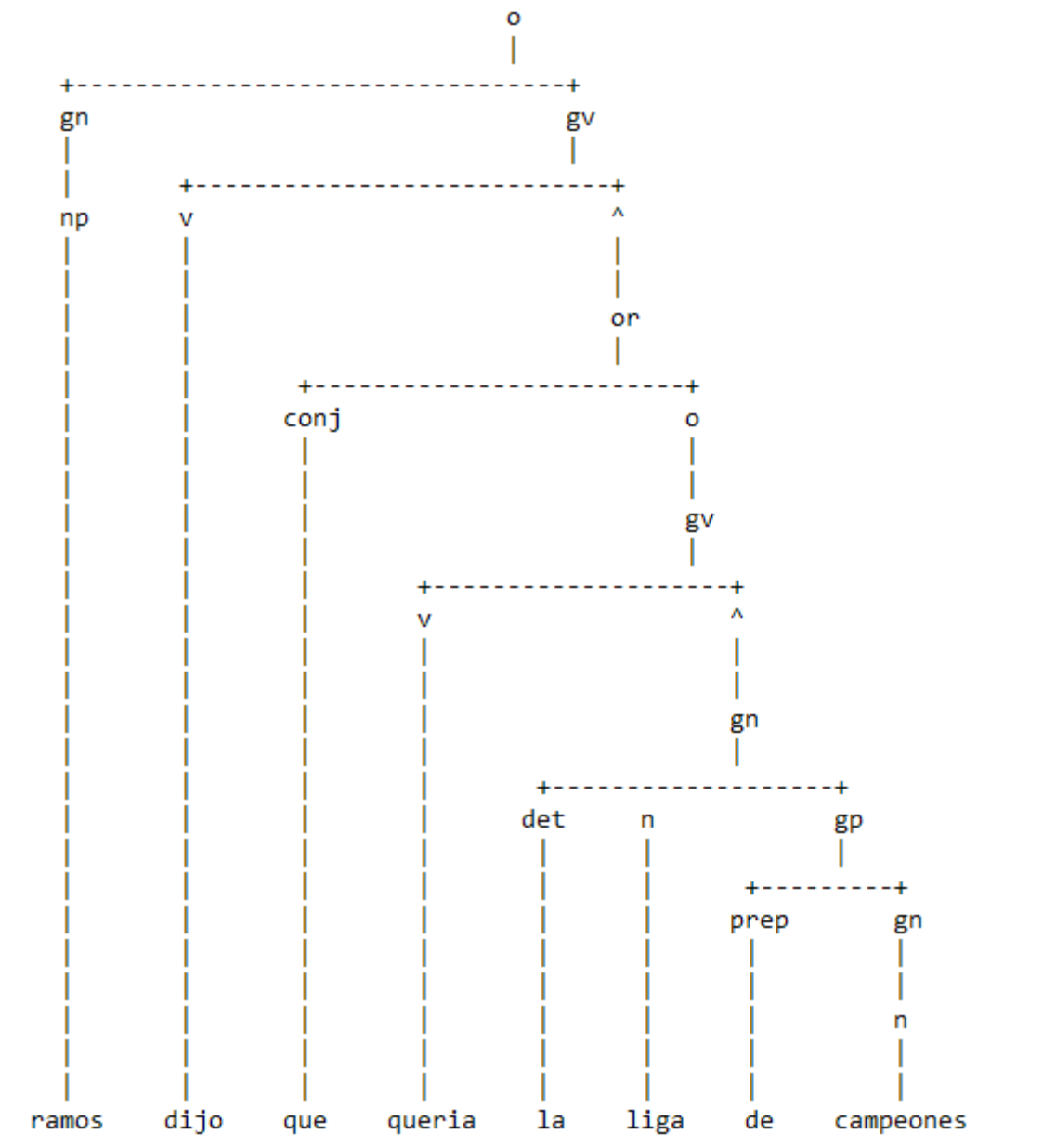
```
oracion(X,[miguel,ha_trabajado,mucho,en,la,practica,y,se_merece,un,die
z],[]),draw(X).
```



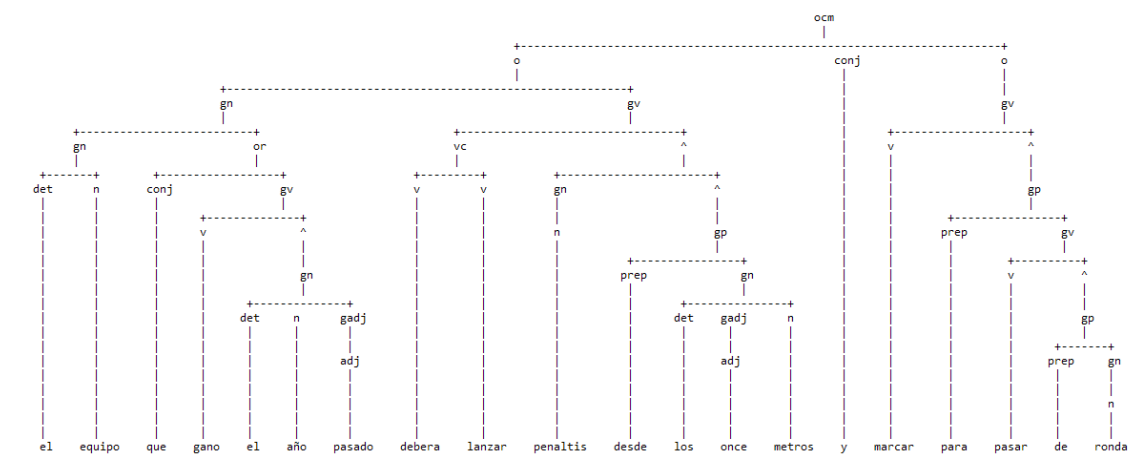
oracion(X,[el,analizador,sintactico,funciona,perfectamente],[]),draw(X).



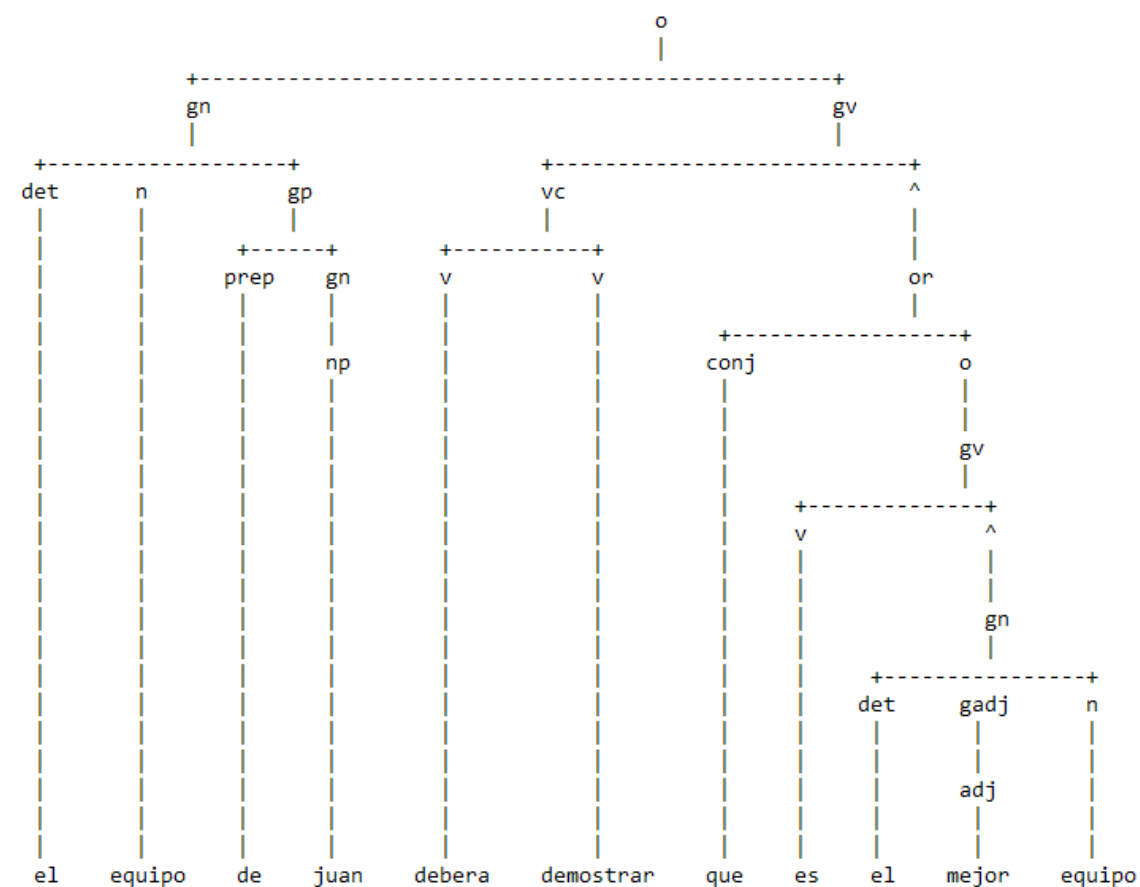
oracion(X,[ramos,dijo,que,queria,la,liga,de,campeones],[]),draw(X).



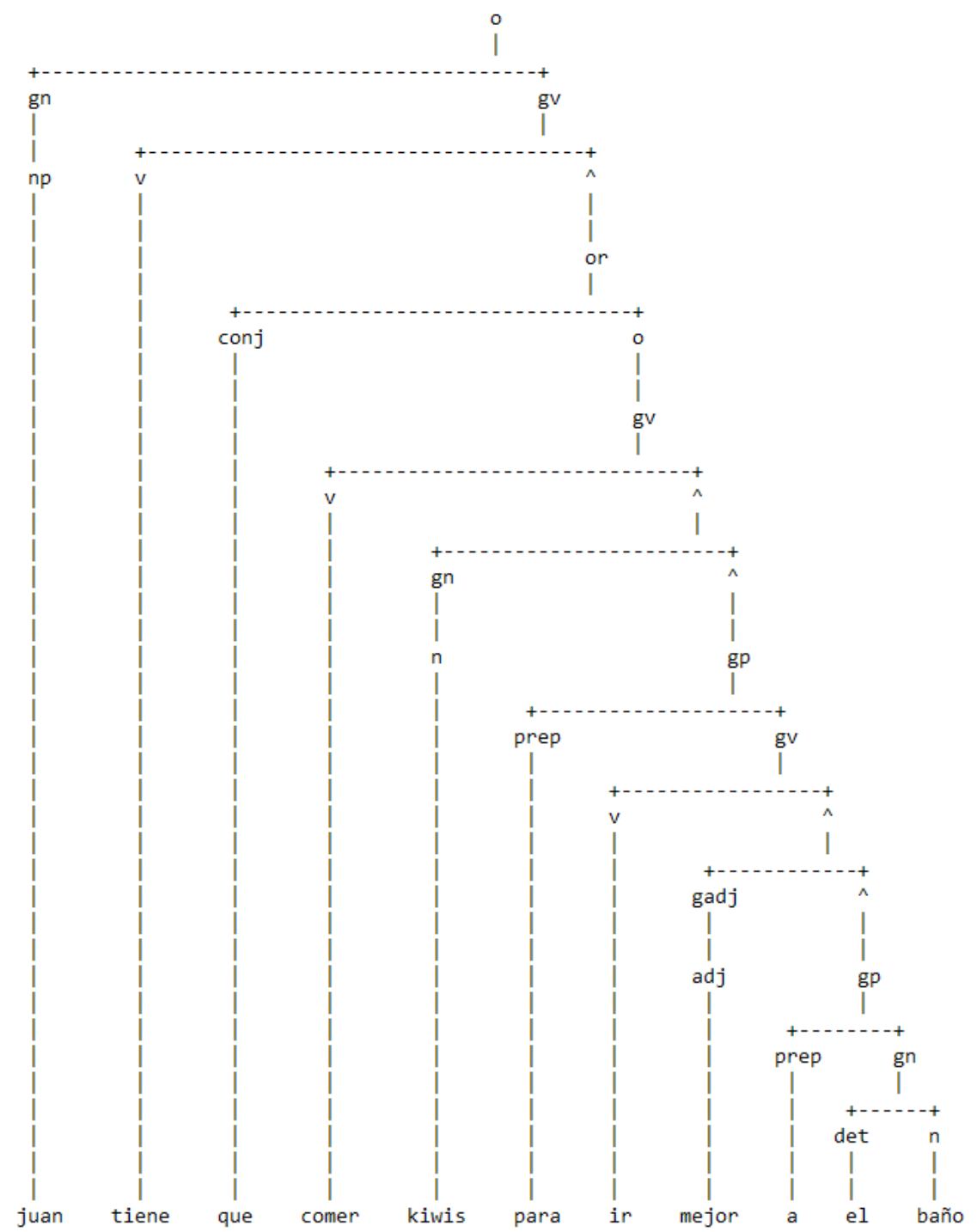
oracion(X,[el,equipo,que,gano,el,año,pasado,debera,lanzar,penaltis,des
de,los,once,metros,y,marcar,para,pasar,de,ronda],[[]],draw(X).



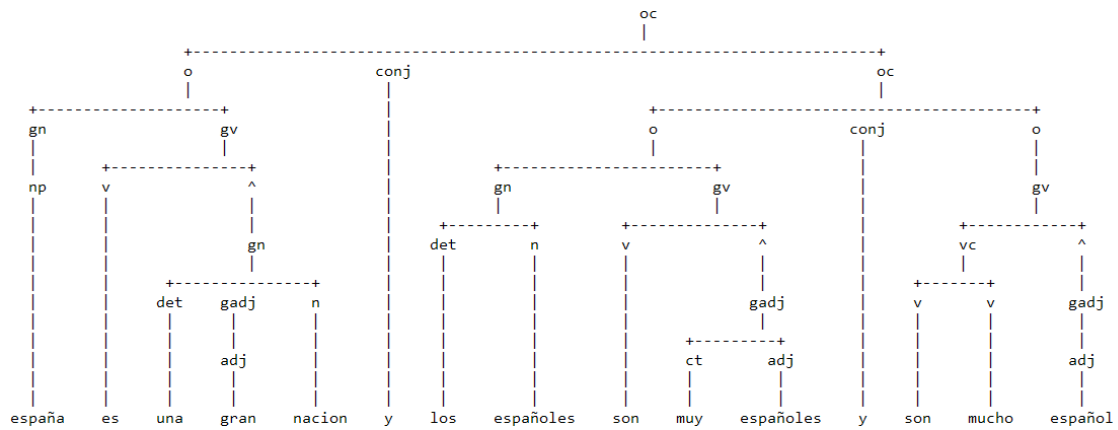
oracion(X,[el, equipo, de, juan, debera, demostrar, que, es, el, mejor, equipo],
[]),draw(X).



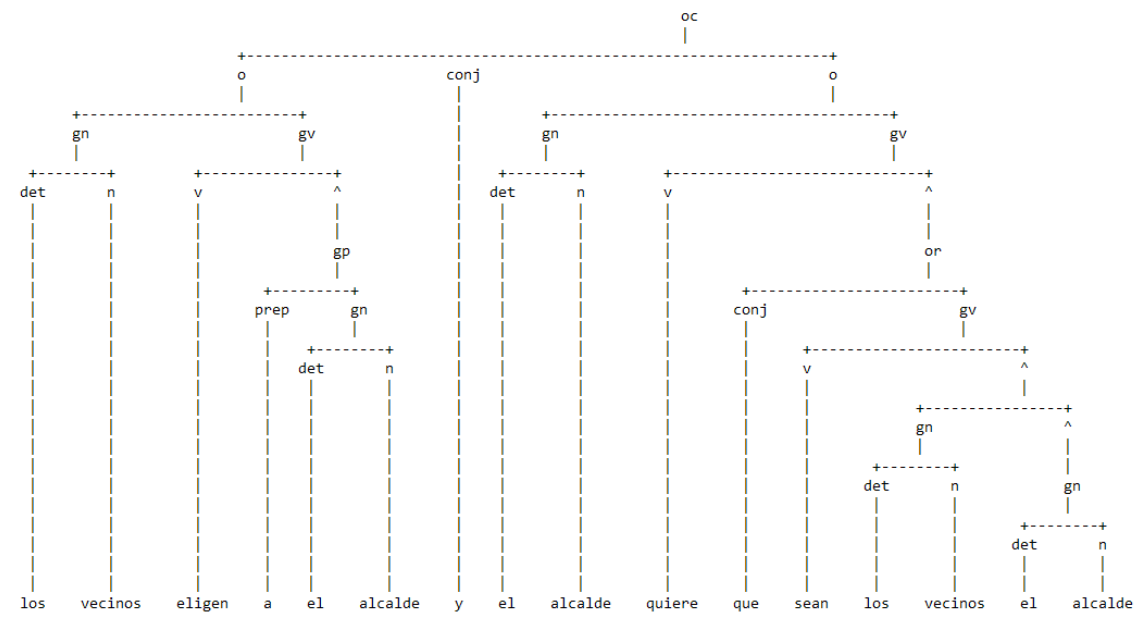
oracion(X,[juan,tiene,que,comer,kiwis,para,ir,mejor,a,el,baño],[]),dra
w(X).



oracion(X,[españa,es,una,gran,nacion,y,los,españoles,son,muy,españoles
 ,y,son,mucho,español],[]),draw(X)



oracion(X,[los,vecinos,eligen,a,el,alcalde,y,el,alcalde,quiere,que,sea
n,los,vecinos,el,alcalde],[]),draw(X).



oracion(X,[la,ceramica,de,talavera,es,no,cosa,menor,y,es,cosa,mayor],[
]),draw(X).

