Uvod u obradu prirodnog jezika

14.1. Probabilističko parsiranje (Probabilistic parsing)

Branko Žitko

prevedeno od: Dan Jurafsky, Chris Manning

Gramatika strukture fraze

$S \rightarrow NP\ VP$

$VP \rightarrow V\ NP$

$VP \rightarrow V\ NP\ PP$

$NP \rightarrow NP\ NP$

$NP \rightarrow NP\ PP$

$NP \rightarrow N$

$NP \rightarrow \varepsilon$

$PP \rightarrow P\ NP$

$N \rightarrow$ primati

$N \rightarrow$ kape

$N \rightarrow$ nose

$N \rightarrow$ glavi

$V \rightarrow$ primati

$V \rightarrow$ kape

$V \rightarrow$ nose

$P \rightarrow$ na

primati kape nose

primati kape na glavi

Gramatike strukture fraze

- Kontekstno neovisne gramatike
  Context Free Grammars (CFG)

- $G = (T, N, S, R)$
  - T – skup terminalnih simbola
  - N – skup neterminalnih simbola
  - S – početni simbol ($S \in N$)
  - R – skup pravila/produkcija oblika $X \rightarrow \gamma$
    $X \in N, \gamma \in (N \cup T)^*$

- Gramatika G generira jezik L

Gramatike strukture fraze u obradi prirodnog jezika

- $G = (T, C, N, S, L, R)$
  - T – skup terminalnih simbola
  - C – skup preterminalnih simbola
  - N – skup neterminalnih simbola
  - S – početni simbol ($S \in N$)
  - L – leksikon, skup elemenata obila $X \rightarrow x$
    $X \in N, x \in T$
  - R – skup pravila/produkcija oblika $X \rightarrow \gamma$
    $X \in N, \gamma \in (N \cup T)^*$

- Gramatika G generira jezik L

CFG

$S \rightarrow NP\ VP$

$VP \rightarrow V\ NP$

$VP \rightarrow V\ NP\ PP$

$NP \rightarrow NP\ NP$

$NP \rightarrow NP\ PP$

$NP \rightarrow N$

$NP \rightarrow \varepsilon$

$PP \rightarrow P\ NP$

$N \rightarrow$ *primati*

$N \rightarrow$ *kape*

$N \rightarrow$ *nose*

$N \rightarrow$ *glavi*

$V \rightarrow$ *primati*

$V \rightarrow$ *kape*

$V \rightarrow$ *nose*

$P \rightarrow$ *na*

primati kape nose

primati kape na glavi

Probabilistička (stohastička) kontekstno neovisna gramatika

Probabilistic CFG (PCFG)

- G = (T, N, S, R, P)
  - T – skup terminalnih simbola
  - N – skup neterminalnih simbola
  - S – početni simbol (S ∈ N)
  - R – skup pravila/produkcija oblika X → γ
    X ∈ N, γ ∈ (N ∪ T)*
  - P – probabilistička funkcija
    P : R→[0, 1]
    $$\forall X \in N, \sum_{X \to \gamma \in R} P(X \to \gamma) = 1$$

- Gramatika G generira model jezika L

$$\sum_{\gamma \in T*} P(\gamma) = 1$$

PCFG

SENT	$\rightarrow$ NP VP	1.0	N	$\rightarrow$ primati	0.5
VP	$\rightarrow$ V NP	0.6	N	$\rightarrow$ kape	0.2
VP	$\rightarrow$ V NP PP	0.4	N	$\rightarrow$ nose	0.2
NP	$\rightarrow$ NP NP	0.1	N	$\rightarrow$ glavi	0.1
NP	$\rightarrow$ NP PP	0.2	V	$\rightarrow$ primati	0.1
NP	$\rightarrow$ N	0.7	V	$\rightarrow$ kape	0.6
~~NP~~	~~$\rightarrow$ ε~~	~~0.0~~	V	$\rightarrow$ nose	0.3
PP	$\rightarrow$ P NP	1.0	S	$\rightarrow$ na	1.0

primati kape nose
primati kape na glavi

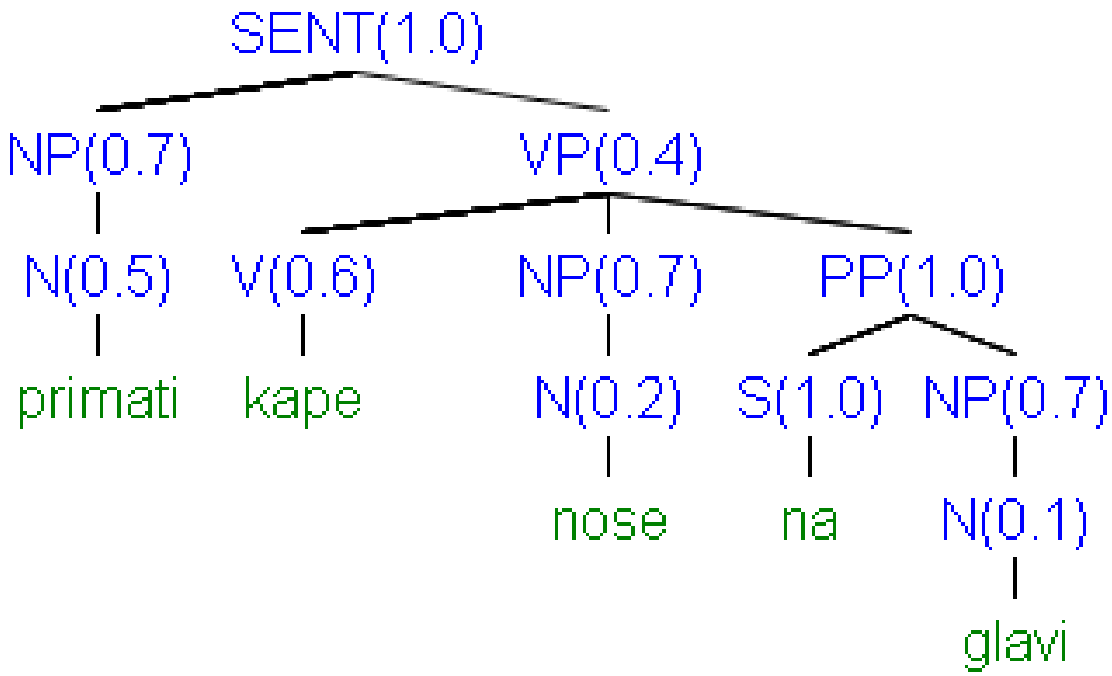
Vjerojatnosti stabala i nizova riječi

- P(t) – vjerojatnost stabla t je umnožak vjerojatnosti pravila korištenih za generiranje stabla
- P(s) – vjerojatnost niza riječi s je suma vjerojatnosti stabala koji generiraju s

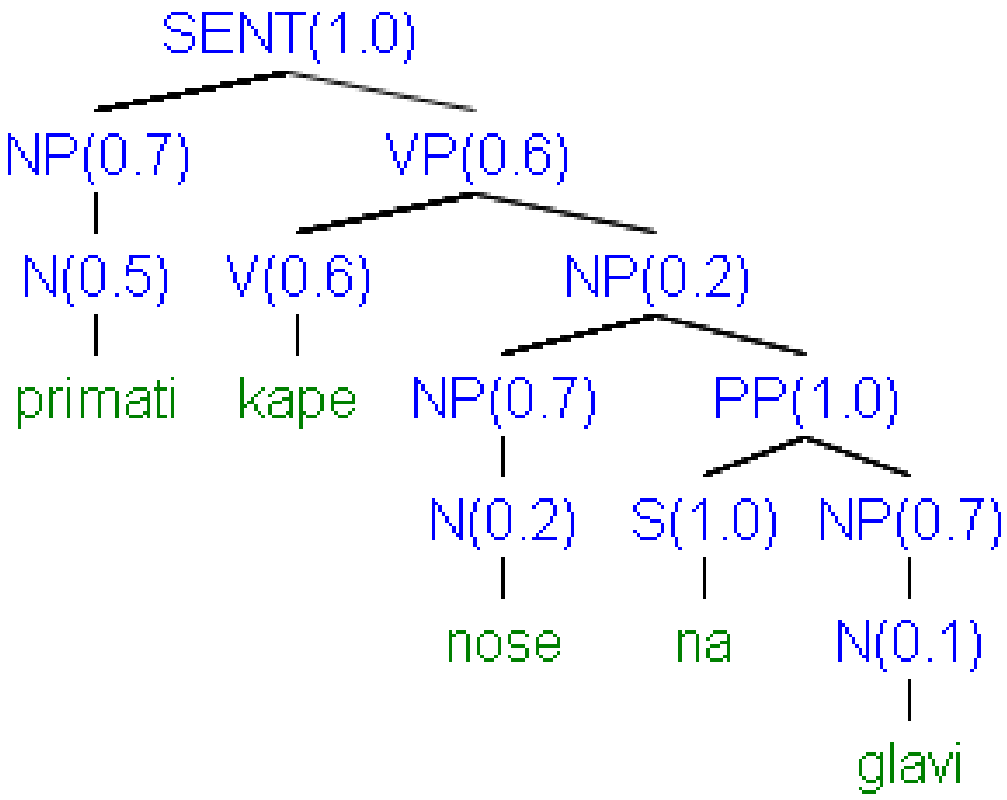
$$P(s) = \Sigma_t P(s, t) \quad \text{gdje je } t \text{ stablo parsiranja od } s$$

$$= \Sigma_t P(t)$$

t_1


SENT(1.0)
NP(0.7)
VP(0.4)
N(0.5)
V(0.6)
NP(0.7)
PP(1.0)
primati
kape
N(0.2)
S(1.0)
NP(0.7)
nose
na
N(0.1)
glavi

t_2


SENT(1.0)
NP(0.7)
VP(0.6)
N(0.5)
V(0.6)
NP(0.2)
primati
kape
NP(0.7)
PP(1.0)
N(0.2)
S(1.0)
NP(0.7)
nose
na
N(0.1)
glavi

Vjerojatnosti stabla i niza riječi

- s = *primati kape nose na glavi*

- $P(t_1) = 1.0 \times 0.7 \times 0.4 \times 0.5 \times 0.6 \times 0.7 \times 1.0 \times 0.2 \times 1.0 \times 0.7 \times 0.1$ Spajanje na glagol
  $= 0.0008232$

- $P(t_2) = 1.0 \times 0.7 \times 0.6 \times 0.5 \times 0.6 \times 0.2 \times 0.7 \times 1.0 \times 0.2 \times 1.0 \times 0.7 \times 0.1$ Spajanje na imenicu
  $= 0.00024696$

- $P(s) = P(t_1) + P(t_2)$
  $= 0.0008232 + 0.00024696$
  $= 0.00107016$

Uvod u obradu prirodnog jezika

14.2. Transformacija gramatike

Branko Žitko

prevedeno od: Dan Jurafsky, Chris Manning

Chomsky normalna forma

- Sva pravila su oblika $X \rightarrow Y\ Z$ ili $X \rightarrow w$
  $X, Y, Z \in N$, $w \in T$
- Transformacija u ovu formu ne mijenja slabo generativno svojstvo CFG
  - Odnosno, prepoznaje isti jezika, ali možda s različitim stablima
- Prazna i unarna pravila se rekurzivno izbacuju
- n-arna pravila se dijele uvođenjem novih neterminala ($n>2$)

CFG

$S \rightarrow NP\ VP$

$VP \rightarrow V\ NP$

$VP \rightarrow V\ NP\ PP$

$NP \rightarrow NP\ NP$

$NP \rightarrow NP\ PP$

$NP \rightarrow N$

$NP \rightarrow \varepsilon$

$PP \rightarrow P\ NP$

$N \rightarrow$ *primati*

$N \rightarrow$ *kape*

$N \rightarrow$ *nose*

$N \rightarrow$ *glavi*

$V \rightarrow$ *primati*

$V \rightarrow$ *kape*

$V \rightarrow$ *nose*

$P \rightarrow$ *na*

Chomsky – eliminacija ε

$S \rightarrow NP\ VP$

$VP \rightarrow V\ NP$

$VP \rightarrow V\ NP\ PP$

$NP \rightarrow NP\ NP$

$NP \rightarrow NP\ PP$

$NP \rightarrow N$

$\cancel{NP \rightarrow \varepsilon}$

$PP \rightarrow P\ NP$

$S \rightarrow NP\ VP$
$S \rightarrow VP$

$VP \rightarrow V\ NP$
$VP \rightarrow NP$

$VP \rightarrow V\ NP\ PP$
$VP \rightarrow V\ PP$

$NP \rightarrow NP\ NP$
$NP \rightarrow NP$

$NP \rightarrow NP\ PP$
$NP \rightarrow PP$

$PP \rightarrow NP\ PP$
$PP \rightarrow P$

$N \rightarrow$ *primati*

$N \rightarrow$ *kape*

$N \rightarrow$ *nose*

$N \rightarrow$ *glavi*

$V \rightarrow$ *primati*

$V \rightarrow$ *kape*

$V \rightarrow$ *nose*

$P \rightarrow$ *na*

Chomsky – eliminacija unarnih pravila

S → NP VP

~~S → VP~~

VP → V NP

VP → V

VP → V NP PP

VP → V PP

NP → NP NP

NP → NP

NP → NP PP

NP → PP

NP → N

PP → P NP

PP → P

S → V NP
S → V
S → V NP PP
S → V PP

N → *primati*

N → *kape*

N → *nose*

N → *glavi*

V → *primati*

V → *kape*

V → *nose*

P → *na*

Chomsky – eliminacija unarnih pravila

S → NP VP
VP → V NP
S → V NP
VP → V
~~S → V~~
VP → V NP PP
S → V NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP
NP → NP PP
NP → PP
NP → N
PP → P NP
PP → P

S → *primati*
S → *kape*
S → *nose*

N → *primati*
N → *kape*
N → *nose*
N → *glavi*
V → *primati*
V → *kape*
V → *nose*
P → *na*

Chomsky – eliminacija unarnih pravila

S → NP VP
VP → V NP
S → V NP
~~VP → V~~
VP → V NP PP
S → V NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP
NP → NP PP
NP → PP
NP → N
PP → P NP
PP → P

VP → *primati*
VP → *kape*
VP → *nose*

N → *primati*
N → *kape*
N → *nose*
N → *glavi*
V → *primati*
S → *primati*
V → *kape*
S → *kape*
V → *nose*
S → *nose*
P → *na*

Chomsky – eliminacija unarnih pravila

$S \rightarrow NP\ VP$
$VP \rightarrow V\ NP$
$S \rightarrow V\ NP$
$VP \rightarrow V\ NP\ PP$
$S \rightarrow V\ NP\ PP$
$VP \rightarrow V\ PP$
$S \rightarrow V\ PP$
$NP \rightarrow NP\ NP$
~~$NP \rightarrow NP$~~
$NP \rightarrow NP\ PP$
$NP \rightarrow PP$
$NP \rightarrow N$
$PP \rightarrow P\ NP$
$PP \rightarrow P$

$N \rightarrow$ *primati*
$N \rightarrow$ *kape*
$N \rightarrow$ *nose*
$N \rightarrow$ *glavi*
$V \rightarrow$ *primati*
$S \rightarrow$ *primati*
$VP \rightarrow$ *primati*
$V \rightarrow$ *kape*
$S \rightarrow$ *kape*
$VP \rightarrow$ *kape*
$V \rightarrow$ *nose*
$S \rightarrow$ *nose*
$VP \rightarrow$ *nose*
$P \rightarrow$ *na*

Chomsky – eliminacija unarnih pravila

S → NP VP
VP → V NP
S → V NP
VP → V NP PP
S → V NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
~~NP → PP~~
NP → N
PP → P NP
PP → P

PP → P NP
PP → P

N → *primati*
N → *kape*
N → *nose*
N → *glavi*
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*

Chomsky – eliminacija unarnih pravila

S → NP VP
VP → V NP
S → V NP
VP → V NP PP
S → V NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
~~NP → N~~
PP → P NP
NP → P NP
PP → P
NP → P

NP → *primati*
NP → *kape*
NP → *nose*
NP → *glavi*

~~N → *primati*~~
~~N → *kape*~~
~~N → *nose*~~
~~N → *glavi*~~
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*

Chomsky – eliminacija unarnih pravila

S → NP VP
VP → V NP
S → V NP
VP → V NP PP
S → V NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
PP → P NP
NP → P NP
~~PP → P~~
NP → P

NP → *primati*
NP → *kape*
NP → *nose*
NP → *glavi*
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*

P → *na*
PP → *na*

Chomsky – eliminacija unarnih pravila

S → NP VP
VP → V NP
S → V NP
VP → V NP PP
S → V NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
PP → P NP
NP → P NP
~~NP → P~~

NP → *primati*
NP → *kape*
NP → *nose*
NP → *glavi*
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*
PP → *na*

NP → *na*

Chomsky – binarizacija

S → NP VP
VP → V NP
S → V NP
~~VP → V NP PP~~
~~S → V NP PP~~
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
PP → P NP
NP → P NP

VP → V @VP-V
@VP-V → NP PP

S → V @S-V
@S-V → NP PP

NP → *primati*
NP → *kape*
NP → *nose*
NP → *glavi*
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*
PP → *na*
NP → *na*

Chomsky normalna forma

S → NP VP
VP → V NP
S → V NP
VP → V @VP-V
@VP-V → NP PP
S → V @S-V
@S-V → NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
PP → P NP
NP → P NP

NP → *primati*
NP → *kape*
NP → *nose*
NP → *glavi*
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*
PP → *na*
NP → *na*

CFG i Chomsky normalna forma

S → NP VP
VP → V NP
VP → V NP PP
NP → NP NP
NP → NP PP
NP → N
NP → ε
PP → P NP

N → *primati*
N → *kape*
N → *nose*
N → *glavi*
V → *primati*
V → *kape*
V → *nose*
P → *na*

S → NP VP
VP → V NP
S → V NP
VP → V @VP-V
@VP-V → NP PP
S → V @S-V
@S-V → NP PP
VP → V PP
S → V PP
NP → NP NP
NP → NP PP
PP → P NP
NP → P NP

NP → *primati*
NP → *kape*
NP → *nose*
NP → *glavi*
V → *primati*
S → *primati*
VP → *primati*
V → *kape*
S → *kape*
VP → *kape*
V → *nose*
S → *nose*
VP → *nose*
P → *na*
PP → *na*
NP → *na*

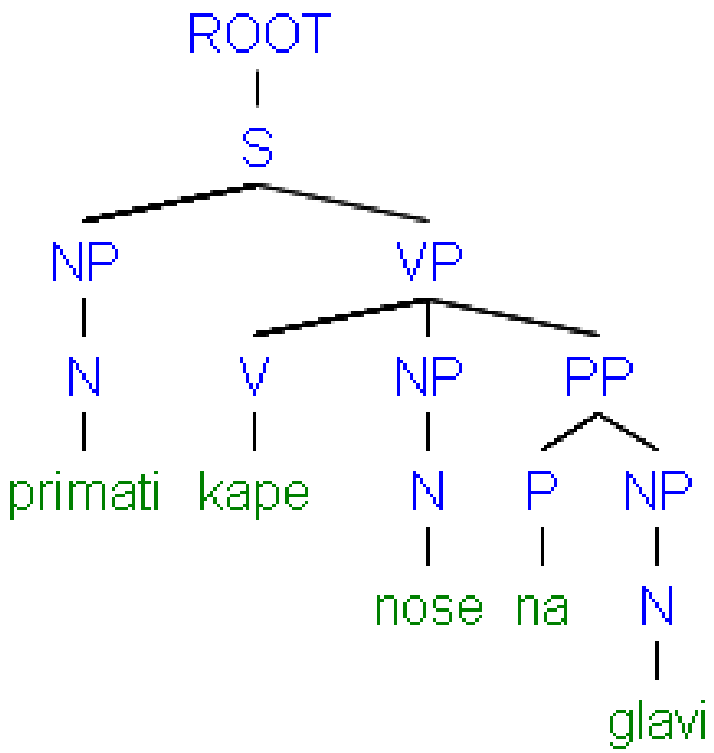
Chomsky normalna forma

- Transformacija radi efikasnog parsiranja
- S pažljivim izborom neterminala moguće je rekonstruirati ista stabla detransformacijom
- Chomsky normalizacija nije lagana
  - rekonstrukcija n-arnih pravila je laka
  - rekonstrukcija unarnih i praznih pravila je složenija

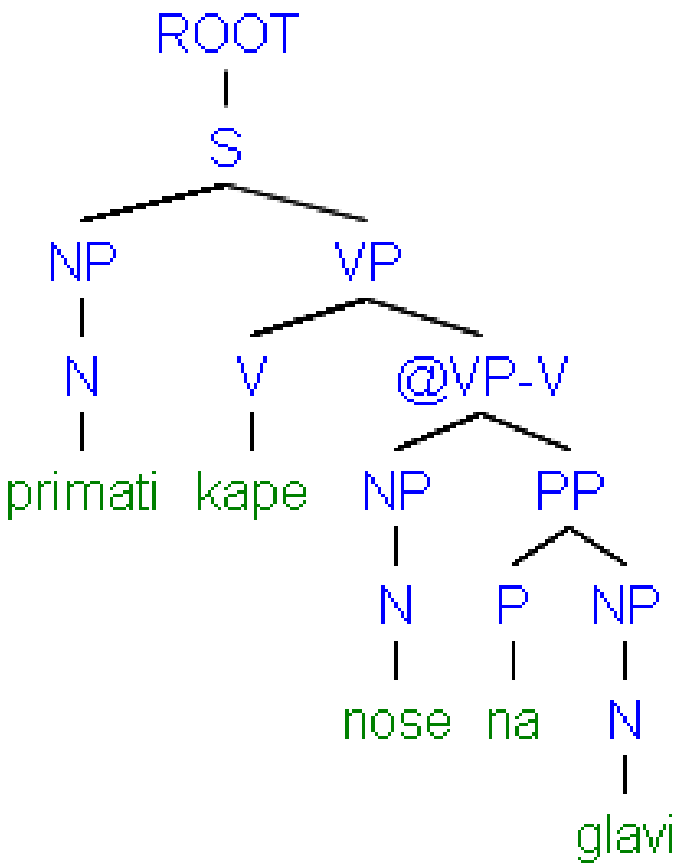
- **Binarizacija** je krucijalna za $O(n^3)$ parsiranje CFG

Primjer: prije i poslije binarizacije

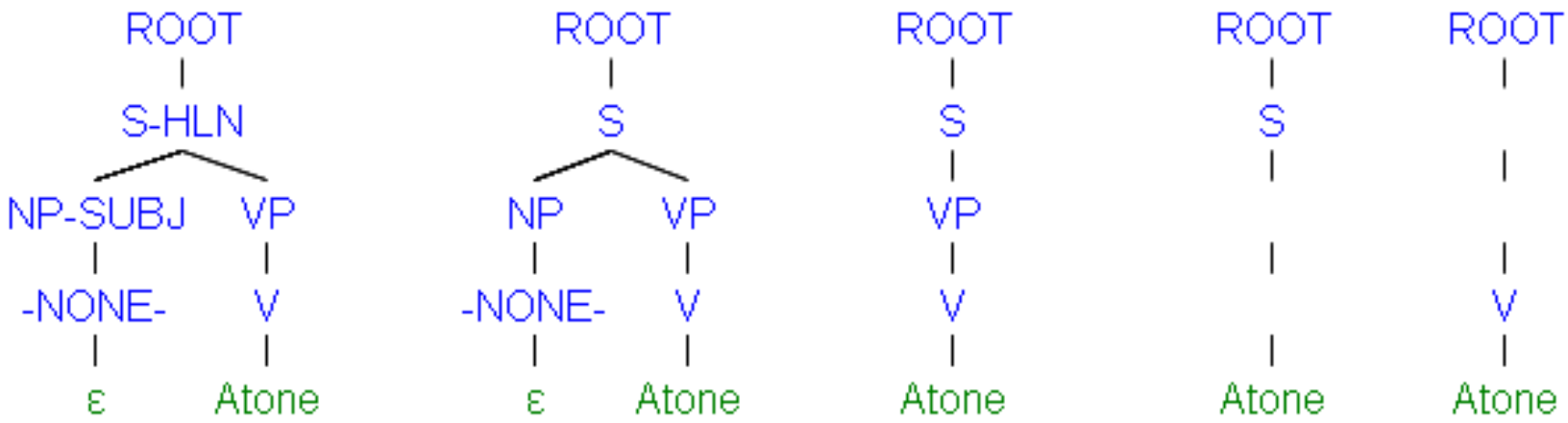
CFG



Normalizirani CFG



Banka stabala: unarna i prazna pravila



PTB Stablo

Bez funkcijskih oznaka

Bez praznih pravila

Visoko

Nisko

Bez unarnih pravila

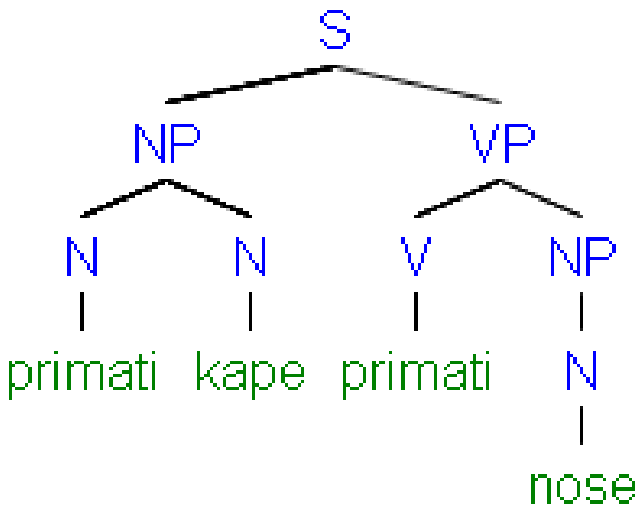
Uvod u obradu prirodnog jezika

14.3. CKY parsiranje

Branko Žitko

prevedeno od: Dan Jurafsky, Chris Manning

Strukturno parsiranje



PCFG

Vjerojatnost pravila θ_i

$S \rightarrow NP\ VP \quad \theta_1$

$VP \rightarrow V\ NP \quad \theta_2$

…

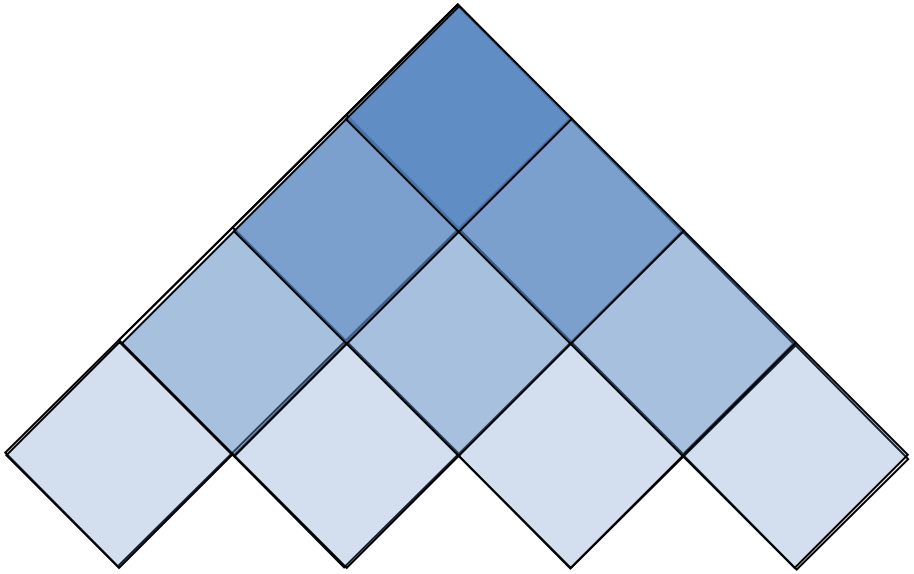
$N \rightarrow$ *primati* $\quad \theta_{42}$

$N \rightarrow$ *kape* $\quad \theta_{43}$

$V \rightarrow$ *primati* $\quad \theta_{44}$

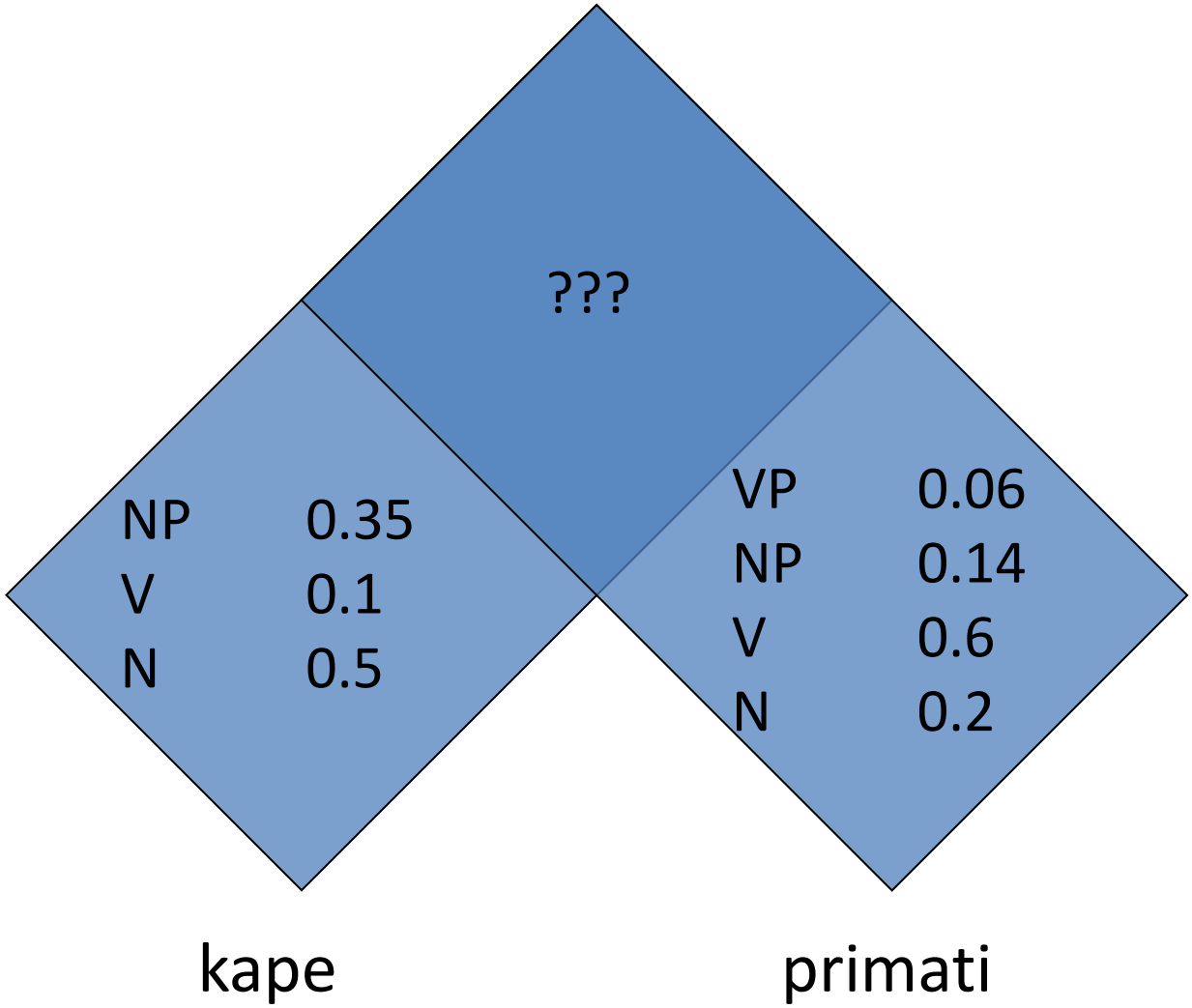
…

Cocke-Kasami-Younger (CKY) parsiranje



kape primati kape nose

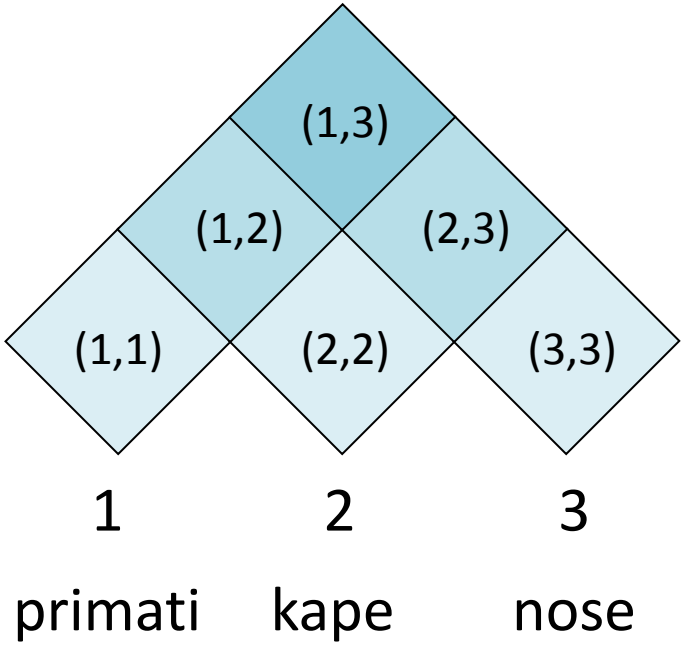
Viterbi (maksimalno bodovanje)

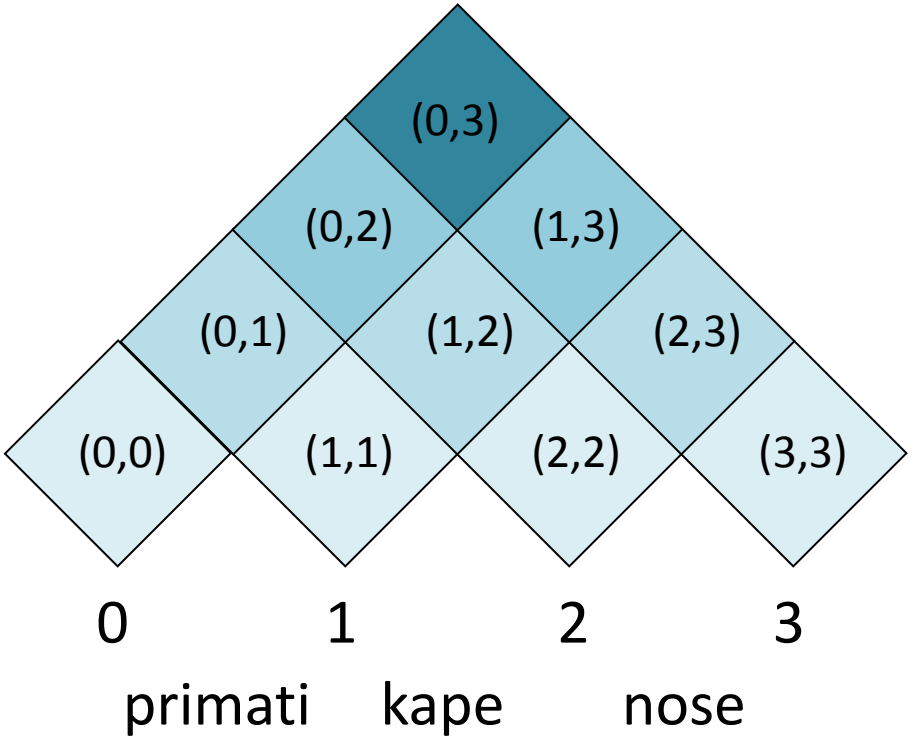


S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

Prošireno CKY parsiranje

- Unarna pravila se mogu uključiti u algoritam
  - malo neredno, ali ne povećava složenost algoritma
- Prazna pravila se mogu uključiti u algoritam
  - korištenje praznih ćelija
  - ne povećava složenost algoritma; slično kao unarna pravila





Prošireno CKY parsiranje

- Binarizacija je vitalna
  - bez binarizacije se ne dobiva kubično vrijeme parsiranja u odnosu na duljinu rečenice i broja neterminala u gramatici
  - Binarizacija može biti eksplicitna ili implicitna kod rada algoritma (kao Earley-ev algoritam), ali je uvijek prisutna

CKY algoritam

```
function CKY(rijeci, gramatika)
    # inicijalizacija
    bod = realna matrica dimenzije |rijeci|+1 x |rijeci|+1 x |neterminali|
    nazad = matrica parova dimenzije |rijeci|+1 x |rijeci|+1 x |neterminali|

    # prvi red
    for pocetak = 0 to |rijeci|-1 do
        kraj = pocetak + 1
        for A -> rijeci[pocetak] in gramatika do
            bod[pocetak][kraj][A] = P(A -> rijeci[pocetak])

        # unarna pravila za prvi red
        dodan = True
        while dodan do
            dodan = False
            for A -> B in gramatika do
                if bod[pocetak][kraj][B] > 0 then
                    prob = P(A->B) * bod[pocetak][kraj][B]
                    if prob > bod[pocetak][kraj][A] then
                        bod[pocetak][kraj][A] = prob
                        nazad[pocetak][kraj][A] = B
                        dodan = True
```

CKY algoritam

```
# ostali redovi
for raspon = 2 to |rijeci| do
    for pocetak = 0 to |rijeci|-raspon do
        kraj = pocetak + raspon
        for podjela = pocetak+1 to kraj-1 do
            for A -> B C in gramatika do
                prob = bod[pocetak][podjela][B] * bod[podjela][kraj][C] *
                       P(A -> B C)
                if prob > bod[pocetak][kraj][A] then
                    bod[pocetak][kraj][A] = prob
                    nazad[pocetak][kraj][A] = (podjela, B, C)

        # unarna pravila za ostale redove
        dodan = True
        while dodan do
            dodan = False
            for A -> B in gramatika do
                prob = P(A -> B) * bod[pocetak][kraj][B]
                if prob > bod[pocetak][kraj][A] then
                    bod[pocetak][kraj][A] = prob
                    nazad[pocetak][kraj][A] = B
                    dodan = True

return NapraviStablo(bod, nazad)
```

Uvod u obradu prirodnog jezika

14.4. CKY parsiranje: primjer

Branko Žitko

prevedeno od: Dan Jurafsky, Chris Manning

Binarna gramatika bez praznih pravila

S → NP VP	0.9	N → *primati*	0.5
S → VP	0.1	N → *kape*	0.2
VP → V NP	0.5	N → *nose*	0.2
VP → V	0.1	N → *glavi*	0.1
VP → V @VP_V	0.3	V → *primati*	0.1
VP → V PP	0.1	V → *kape*	0.6
@VP_V → NP PP	1.0	V → *nose*	0.3
NP → NP NP	0.1	P → *na*	1.0
NP → NP PP	0.2		
NP → N	0.7		
PP → P NP	1.0		

CKY

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape	1	primati	2	kape	3	nose	4
0	bod[0][1]		bod[0][2]		bod[0][3]		bod[0][4]	
1			bod[1][2]		bod[1][3]		bod[1][4]	
2					bod[2][3]		bod[2][4]	
3							bod[3][4]	
4								

CKY – leksička pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape	primati	kape	nose
0		1	2	3
1	N → kape 0.2 V → kape 0.6			
2		N → primati 0.5 V → primati 0.1		
3			N → kape 0.2 V → kape 0.6	
4				N → nose 0.1 V → nose 0.3

```
# prvi red
for pocetak = 0 to |rijeci|-1 do
    kraj = pocetak + 1
    for A -> rijeci[pocetak] do
        bod[pocetak][kraj][A] = P(A -> rijeci[pocetak])
```

CKY – unarna pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape	primati	kape	nose
0	N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006			
1		N → primati 0.5 V → primati 0.1 NP → N 0.35 VP → V 0.01 S → VP 0.001		
2			N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006	
				N → nose 0.1 V → nose 0.3 NP → N 0.14 VP → V 0.03 S → VP 0.003

```
# unarna pravila za prvi red
dodan = True
while dodan do
    dodan = False
    for A -> B in gramatika do
        if bod[pocetak][kraj][B] > 0 then
            prob = P(A->B) * bod[pocetak][kraj][B]
            if prob > bod[pocetak][kraj][A] then
                bod[pocetak][kraj][A] = prob
                nazad[pocetak][kraj][A] = B
                dodan = True
```

CKY – binarna pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape	1	primati	2	kape	3	nose	4
0	N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006		NP → NP NP 0.0049 VP → V NP 0.105 S → NP VP 0.00126					
1			N → primati 0.5 V → primati 0.1 NP → N 0.35 VP → V 0.01 S → VP 0.001		NP → NP NP 0.0049 VP → V NP 0.007 S → NP VP 0.0189			
2					N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006		NP → NP NP 0.00196 VP → V NP 0.042 S → NP VP 0.00378	
3							N → nose 0.1 V → nose 0.3 NP → N 0.14 VP → V 0.03 S → VP 0.003	
4								

```
prob = bod[pocetak][podjela][B] * bod[podjela][kraj][C] * P(A -> B C)
if prob > bod[pocetak][kraj][A] then
    bod[pocetak][kraj][A] = prob
    nazad[pocetak][kraj][A] = (podjela, B, C)
```

CKY – unarna pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape 1	primati 2	kape 3	nose 4
0	N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006	NP → NP NP 0.0049 VP → V NP 0.105 S → VP 0.0105		
1		N → primati 0.5 V → primati 0.1 NP → N 0.35 VP → V 0.01 S → VP 0.001	NP → NP NP 0.0049 VP → V NP 0.007 S → NP VP 0.0189	
2			N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006	NP → NP NP 0.00196 VP → V NP 0.042 S → VP 0.0042
				N → nose 0.1 V → nose 0.3 NP → N 0.14 VP → V 0.03 S → VP 0.003

```
# unarna pravila za ostale redove
dodan = True
while dodan do
    dodan = False
    for A -> B in gramatika do
        prob = P(A -> B) * bod[pocetak][kraj][B]
        if prob > bod[pocetak][kraj][A] then
            bod[pocetak][kraj][A] = prob
            nazad[pocetak][kraj][A] = B
            dodan = True
```

CKY – binarna i unarna pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape 1	primati 2	kape 3	nose 4
0	N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006	NP → NP NP 0.0049 VP → V NP 0.105 S → VP 0.0105	NP → NP NP 0.0000686 VP → V NP 0.00147 S → NP VP 0.000882	
1		N → primati 0.5 V → primati 0.1 NP → N 0.35 VP → V 0.01 S → VP 0.001	NP → NP NP 0.0049 VP → V NP 0.007 S → NP VP 0.0189	
2			N → kape 0.2 V → kape 0.6 NP → N 0.14 0.06 0.006	NP → NP NP 0.00196 VP → V NP 0.042 S → VP 0.0042
				N → nose 0.1 V → nose 0.3 NP → N 0.14 VP → V 0.03 S → VP 0.003

```
# ostali redovi
for raspon = 2 to |rijeci| do
   for pocetak = 0 to |rijeci|-raspon do
      kraj = pocetak + raspon
      for podjela = pocetak+1 to kraj-1 do
         for A -> B C in gramatika do
            prob = bod[pocetak][podjela][B] * bod[podjela][kraj][C]
                   * P(A -> B C)
            if prob > bod[pocetak][kraj][A] then
               bod[pocetak][kraj][A] = prob
               nazad[pocetak][kraj][A] = (podjela, B, C)
```

CKY – binarna i unarna pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape 1	primati 2	kape 3	nose 4
0	N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006	NP → NP NP 0.0049 VP → V NP 0.105 S → VP 0.0105	NP → NP NP 0.0000686 VP → V NP 0.00147 S → NP VP 0.000882	
1		N → primati 0.5 V → primati 0.1 NP → N 0.35 VP → V 0.01 S → VP 0.001	NP → NP NP 0.0049 VP → V NP 0.007 S → NP VP 0.0189	NP → NP NP 0.0000686 VP → V NP 0.000098 S → NP VP 0.01323
2			N → kape 0.2 V → kape 0.6 NP → N 0.14 0.06 0.006	NP → NP NP 0.00196 VP → V NP 0.042 S → VP 0.0042
				N → nose 0.1 V → nose 0.3 NP → N 0.14 VP → V 0.03 S → VP 0.003

```
# ostali redovi
for raspon = 2 to |rijeci| do
   for pocetak = 0 to |rijeci|-raspon do
      kraj = pocetak + raspon
      for podjela = pocetak+1 to kraj-1 do
         for A -> B C in gramatika do
            prob = bod[pocetak][podjela][B] * bod[podjela][kraj][C]
                   * P(A -> B C)
            if prob > bod[pocetak][kraj][A] then
               bod[pocetak][kraj][A] = prob
               nazad[pocetak][kraj][A] = (podjela, B, C)
```

CKY – binarna i unarna pravila

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape 1	primati 2	kape 3	nose 4
0	N → kape 0.2 V → kape 0.6 NP → N 0.14 VP → V 0.06 S → VP 0.006	NP → NP NP 0.0049 VP → V NP 0.105 S → VP 0.0105	NP → NP NP 0.0000686 VP → V NP 0.00147 S → NP VP 0.000882	NP → NP NP 0.0000009604 VP → V NP 0.00002058 S → NP VP 0.00018522
1		N → primati 0.5 V → primati 0.1 NP → N 0.35 VP → V 0.01 S → VP 0.001	NP → NP NP 0.0049 VP → V NP 0.007 S → NP VP 0.0189	NP → NP NP 0.0000686 VP → V NP 0.000098 S → NP VP 0.01323
2			N → kape 0.2 V → kape 0.6 NP → N 0.14 0.06 0.006	NP → NP NP 0.00196 VP → V NP 0.042 S → VP 0.0042
				N → nose 0.1 V → nose 0.3 NP → N 0.14 VP → V 0.03 S → VP 0.003

```
# ostali redovi
for raspon = 2 to |rijeci| do
   for pocetak = 0 to |rijeci|-raspon do
      kraj = pocetak + raspon
      for podjela = pocetak+1 to kraj-1 do
         for A -> B C in gramatika do
            prob = bod[pocetak][podjela][B] * bod[podjela][kraj][C]
                   * P(A -> B C)
            if prob > bod[pocetak][kraj][A] then
               bod[pocetak][kraj][A] = prob
               nazad[pocetak][kraj][A] = (podjela, B, C)
```

CKY – vraćanje unatrag

S → NP VP	0.9
S → VP	0.1
VP → V NP	0.5
VP → V	0.1
VP → V @VP_V	0.3
VP → V PP	0.1
@VP_V → NP PP	1.0
NP → NP NP	0.1
NP → NP PP	0.2
NP → N	0.7
PP → P NP	1.0

N → *primati*	0.5
N → *kape*	0.2
N → *nose*	0.2
N → *glavi*	0.1
V → *primati*	0.1
V → *kape*	0.6
V → *nose*	0.3
P → *na*	1.0

	kape	primati	kape	nose
0–1	N → kape V → kape NP → N VP → V S → VP	NP → NP NP 1 VP → V NP S → VP	NP → NP NP VP → V NP S → NP VP	NP → NP NP VP → V NP S → NP VP 2
1–2		N → primati V → primati NP → N VP → V S → VP	NP → NP NP VP → V NP S → NP VP	NP → NP NP VP → V NP S → NP VP
2–3			N → kape V → kape NP → N VP → V S → VP	NP → NP NP VP → V NP 3 S → VP
3–4				N → nose V → nose NP → N VP → V S → VP

Uvod u obradu prirodnog jezika

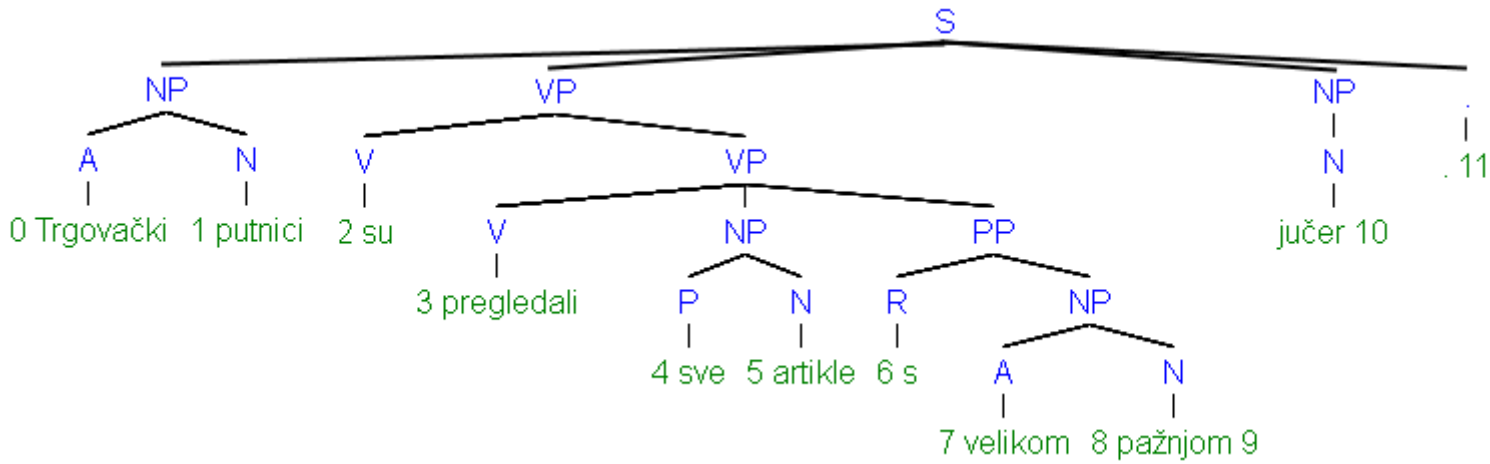
14.5. Evaluacija strukturnog parsiranja

Branko Žitko

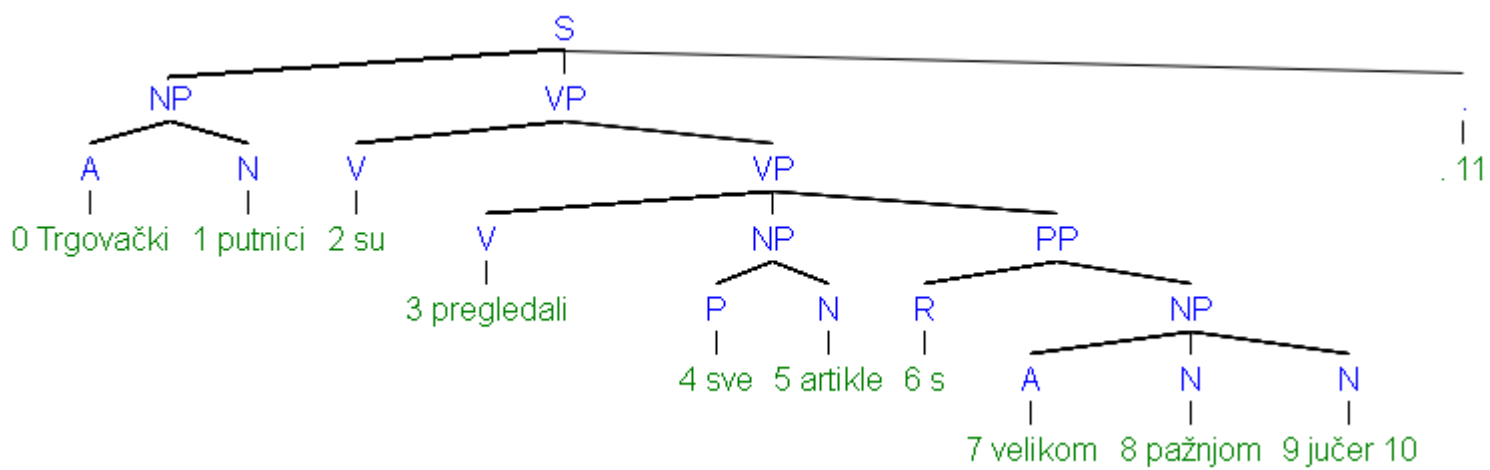
prevedeno od: Dan Jurafsky, Chris Manning

Evaluacija strukturnog parsiranja

Zlatni standard: **S(0:11) NP(0:2)** VP(2:9) VP(3:9) **NP(4:6)** PP(6:9) NP(7:9) NP(9:10)



Kandidat: **S(0:11) NP(0:2)** VP(2:10) VP(3:10) **NP(4:6)** PP(6:10) NP(7:10)



Evaluacija strukturnog parsiranja

Zlatni standard:

S(0:11) NP(0:2) VP(2:9) VP(3:9) **NP(4:6)** PP(6:9) NP(7:9) NP(9:10)

Kandidat:

S(0:11) NP(0:2) VP(2:10) VP(3:10) **NP(4:6)** PP(6:10) NP(7:10)

Preciznost oznake (PO):	3/7 = 42.9%
Odziv oznake (OO):	3/8 = 37.5%
F1 oznake:	40%
POS točnost:	11/11 = 100%

Koliko su dobre PCFG?

- Točnost parsiranja Penn WSJ: oko 73% F1
- Robusno
  - Obično prihvaća sve, ali s malom vjerojatnošću
- Parcijalno rješenje za višeznačnost gramatike
  - PCFG daje neke ideje vjerojatnosti parsiranja
  - ali ne toliko dobre jer su pretpostavke nezavisnosti previše jake
- Daju probabilistički model jezika
  - ali kod jednostavnih slučajeva daje lošije rezultate od trigram modela
- Izgleda da je problem PCFG-a u nedostatku leksikalizacije trigram modela