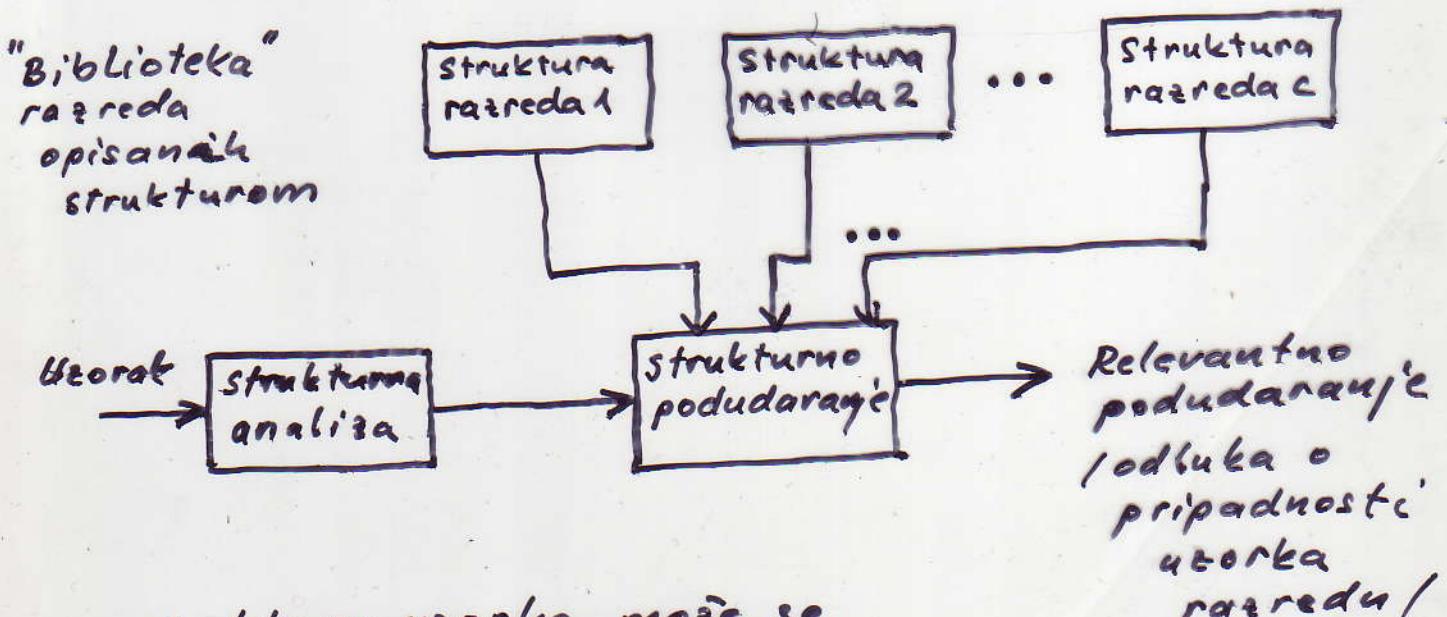


JEZIČNI (SINTAKTIČKI) PRISTUP RASPOZNAVANJU UZORKA

- uporaba formalne jezične teorije
 jezični, sintaktički pristup, lingvistički pristup
 strukturalno raspoznavanje uzorka
 (engl. Structural Pattern Recognition)
 OSNOVNA PREPOSTAVKA: struktura entiteta
 sintaktički pristup koristi strukturu uzorka
 u postupku RASPOZNAVANJA

Analistički pristup \leftrightarrow strukturalni pristup
 / kvantitativna svojstva
 uzorka; u velikoj mjeri
 ignorira se međusobni
 odnos između komponenti
 uzorka /
 / odnos između
 komponenti
 uzorka
 ↓
 usmjeren prema
 slikevno predstavljanim
 uzorcima /

Shematski prikaz sustava
za sintaktičko raspoznavanje:



- struktura uzorka može se kvantificirati!
- za kvantifikaciju strukture rabe se dva pristupa:
 - a) formalne gramatike;
 - b) relacijski opisi (grafovi);

- rečina sintaktskog pristupa raspoznavanja, temelji se na analizi složenih uzoraka uporabom HIERARHIJSKE DEKOMPOZICIJE NA JEDNOSTAVNije UZORKE

OSNOVNI POJMOVI IZ TEORIJE FORMALNIH JEZIKA

Abeceda (engl. Alphabet) - konacan skup simbola (znakova)

Recenica (engl. Sentence) - definirana nad nekom abecedom - niz konacne duljine sastavljen od simbola abecede

Primjer: abeceda = {0, 1} niz (string, word)
recenice : {0, 1, 00, 01, 10, 101, ...}

Recenica koja nema simbola (prazna recenica, prazni niz): ϵ (ili s_0 ili λ)

Duljina niza - broj znakova u nizu;
niz s ; $|s|$ - duljina niza s

- za prazan niz vrijedi:

$$x \circ \epsilon = \epsilon \circ x = x$$

$$|\epsilon| = 0$$

' \circ ' - označava
lancanje - konkatenaciju!

- za neku abecedu V rabimo V^* za označavanje svih recenica (nizova) koji su sastavljeni od simbola iz V (uključujući i prazni simbol ϵ)

$$V^+ = V^* - \{\epsilon\} ; V^* = \{\epsilon\} \cup V^+$$

Primjer :

$$V = \{a, b\}$$

$$V^* = \{\epsilon, a, b, ab, aa, bb, \dots\}$$

$$V^+ = \{a, b, ab, aa, bb, \dots\}$$

Jezik - bilo koji skup rečenica (ne nužno konačni skup) definiran nad nekom abecedom

Gramatika 4-tvorka

$$G = (V_N, V_T, P, S)$$

V_N - skup neterminala (varijabli);

V_T - skup terminala (konstanti);

P - skup produkcija (ili producijiskih pravila);

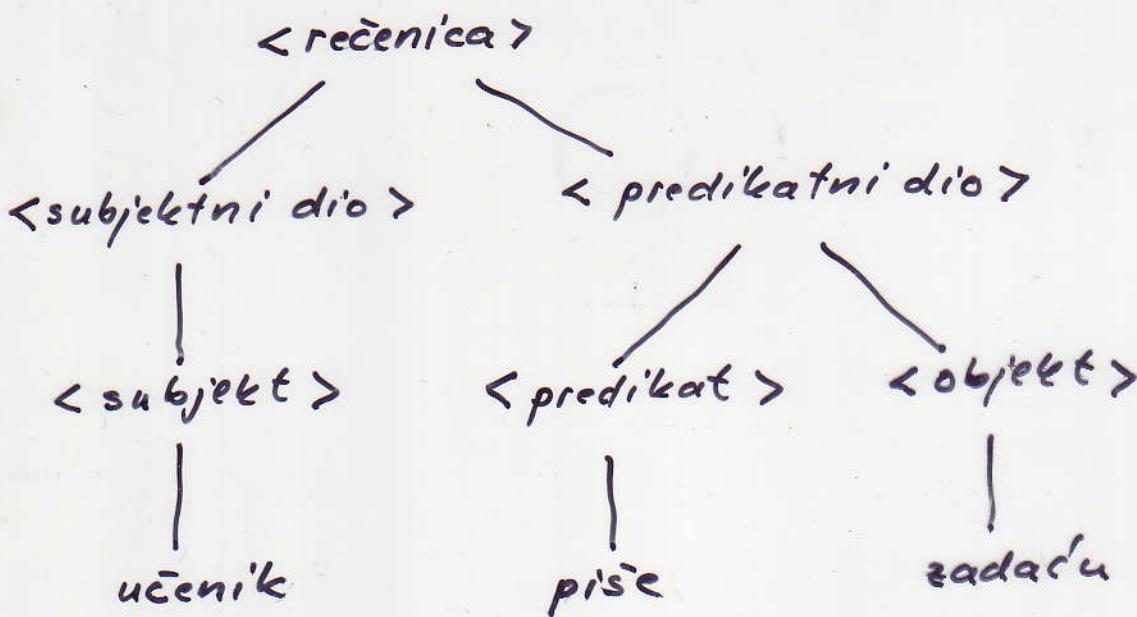
S - početni (startni) simbol (korijen; root)

$$S \in V_N \quad V_N \cap V_T = \emptyset \quad V = V_N \cup V_T$$

Primjer: Usporedba sa gramatikom prirodnog jezika

54

"Učenik piše zadacu."



$$V_N = \{ \langle \text{subjektni dio} \rangle, \langle \text{predikatni dio} \rangle, \langle \text{subjekt} \rangle \\ \langle \text{predikat} \rangle, \langle \text{objekt} \rangle, \langle \text{rečenica} \rangle \dots \}$$

$$V_T = \{ \text{učenik}, \text{piše}, \text{zadacu} \}$$

Produkcijska pravila P

$$\begin{aligned} \langle \text{rečenica} \rangle &\rightarrow \langle \text{subjektni dio} \rangle \langle \text{predikatni dio} \rangle \\ \langle \text{subjektni dio} \rangle &\rightarrow \langle \text{subjekt} \rangle \\ \langle \text{predikatni dio} \rangle &\rightarrow \langle \text{predikat} \rangle \langle \text{objekt} \rangle \\ \langle \text{subjekt} \rangle &\rightarrow \text{učenik} \\ \langle \text{predikat} \rangle &\rightarrow \text{piše} \\ \langle \text{objekt} \rangle &\rightarrow \text{zadacu} \end{aligned}$$

Jezik koji se generira pomoću gramatike G označava se sa $L(G)$ predstavljajući skup nizova koji zadovoljavaju sljedeća dva uvjeta:

- (1) svaki se niz sastoji samo od terminala;
- (2) svaki niz se može izvesti iz startnog simbola S pomoću odgovarajuće primjene produkcijskih pravila iz skupa P ;

Dogovor:

- Neterminale čemo označavati sa: S, A, B, \dots
- Terminalne sa: a, b, c
- Nizove terminala sa: v, w, α, \dots
- Nizove koji se sastoje od terminala i neterminala sa: $\alpha, \beta, \gamma, \delta, \dots$

Skup produkcija P sastoji se od izraza oblike

$$\alpha \rightarrow \beta,$$

gdje je $\alpha \in V^+$ i $\beta \in V^*$; $V = V_N \cup V_T$
 "→" označava zamjenu niza α s nizom β ;
 simbol \xrightarrow{G} označavaće operaciju

oblike

$$y \xrightarrow{G} \gamma \alpha \delta \Rightarrow y \beta \delta$$

\xrightarrow{G} ili (\Rightarrow) pokazuje zamjenu α s β pomoću produkcijskog pravila $\alpha \rightarrow \beta$, pri čemu y i δ se ne mijenjaju;

Primjer:

$$G = (V_N, V_T, P, S)$$

$$V_N = \{ S \}$$

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

$$P = \{ S \rightarrow aSb, S \rightarrow ab \}$$

Ako prvo produkcijuško pravilo uporabimo $m-1$ puta dobivamo

$$\begin{aligned} S &\Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow \\ &\Rightarrow a^4Sb^4 \Rightarrow \dots \Rightarrow a^{m-1}Sb^{m-1} \end{aligned}$$

Uporabom drugog produkcijuškog pravila dobivamo:

$$a^{m-1}Sb^{m-1} \Rightarrow a^m b^m$$

$$L(G) = \{ a^m b^m \mid m \geq 1 \}$$

- jednostravna gramatika će ovog primjera generirati jezik s beskonačnim brojem nizova ili rečenica!

Tipovi gramatika

- sve su gramatike oblike $G = (V_N, V_T, P, S)$ ali se razlikuju po tipu dopuštenih producijuških pravila;

a) Gramatika tipa 0

(slobodna ili neograničena gramatika)

/engl. Free ; Unrestricted /

Produktivska pravila su oblika

$$\alpha \rightarrow \beta ,$$

gdje je α iz V^+ i β niz iz V^*

Primjer: $G = (V_N, V_T, P, S)$

$$V_N = \{S, A, B\} \quad V_T = \{a, b, c\}$$

$$P: S \rightarrow aAbc$$

$$Ab \rightarrow bA$$

$$Ac \rightarrow Bbcc$$

$$bB \rightarrow Bb$$

$$aB \rightarrow aaA$$

$$aB \rightarrow \epsilon$$

$L(G)$ m rečenice $a^n b^{n+2} c^{n+2}$, $n \geq 0$

Npr. da bi se generirao niz

$$\alpha = a^0 b^2 c^2 = bbcc \text{ treba se}$$

$$S \Rightarrow aAbc \Rightarrow abAc \Rightarrow abBbcc \Rightarrow$$

$$\Rightarrow aBbcc \Rightarrow bcc$$

Produktiva tipa $aB \rightarrow \epsilon$ je dopuštena samo za slobodne gramatike, t.zv. 'erasing production'

↪ ne postoje ograničenje :

$$|\alpha| \leq |\beta|$$

- koncept koji dopušta razmatrajuće gramatiku koje generiraju varijacije utorka koje uključuju brisanje poduzoraka;

b) Gramatika tipa 1

(kontekstno osjetljiva; kontekstno zavisna)
(engl. Context-sensitive)

- produkcjska pravila oblike:

$$\alpha_1 A \alpha_2 \rightarrow \alpha_1 \beta \alpha_2,$$

gdje su α_1 i α_2 iz V^* a
 β iz V^+
 A iz V_N

Ova gramatika dopušta zamjenu neterminala A nizom β SAMO kada se A javlja u kontekstu $(\alpha_1 A \alpha_2)$ nizova α_1 i α_2

Primjer: $G = (V_N, V_T, P, S)$

$$V_N = \{S, A, B\} \quad V_T = \{a, b, c\}$$

$$\begin{aligned} P: \quad & S \rightarrow abc \\ & S \rightarrow aAbc \\ & Ab \rightarrow bA \\ & Ac \rightarrow Bbcc \\ & bB \rightarrow Bb \\ & aB \rightarrow aaA \\ & aB \rightarrow aa \end{aligned}$$

c) Gramatika tipa 2

(kontekstno slobodna, kontekstno nesavrsna)
 (engl. context-free)

- produkcije oblike:

$$A \rightarrow \beta$$

$$A \in V_N$$

$$\beta \in V^+$$

- Neterminál A može biti zamjenjen nizom β bez obzira na kontekst u kojem se A pojavljuje;
- Gramatika tipa 2 može generirati niz terminala ili neterminala (ili oboje) jednom produkcijom;
- Dopusćena je i produkcija oblike

$$A \rightarrow \alpha A \beta$$

tav. self-embedding gramatika

VAŽNO: Kontekstno slobodna gramatika "najopisnija" gramatika za koje su razvijeni djelotvorni parseri

Primjer: $G = (V_N, V_T, P, S)$

$$V_N = \{S\} \quad V_T = \{a, b\}$$

$$P: S \rightarrow ab$$

$$S \rightarrow aSb$$

d) Gramatika tipa 3

(Regularna gramatika)

(engl. regular, finite-state)

- Producijačka pravila oblika:

$$A \rightarrow aB \text{ ili } A \rightarrow a$$

A, B iz V_N

a, b iz V_T

Alternativno napisane produkcije su

$$A \rightarrow Ba$$

$$A \rightarrow a$$

Međusobno
se
isključuju!

- regulare gramatike podesne za analizu (parsiranje) konacnim automatom

Primjer: $G = \{V_T, V_N, P, S\}$

$$V_T = \{a, b\} \quad V_N = \{S, A_1, A_2\}$$

$$P: S \rightarrow aA_2$$

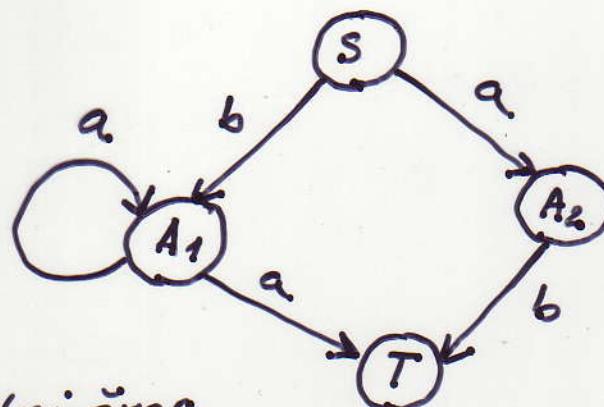
$$S \rightarrow bA_1$$

$$A_1 \rightarrow a$$

$$A_1 \rightarrow aA_1$$

$$A_2 \rightarrow b$$

Grafički prikaz:



T - terminalni čvor

Odnos između gramatika T_i ; $i=0, 1, 2, 3$

Vrijedi:

$$L(T_3) \subset L(T_2) \subset L(T_1) \subset L(T_0)$$

Tip gramatike

$T_0 \quad T_1 \quad T_2 \quad T_3$

$$L(T_0) \supset L(T_1) \supset L(T_2) \supset L(T_3) \quad \text{Generirani jezik}$$

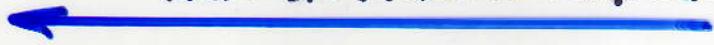
Rast ograničenja za produkcija pravila



Rast opisnih sposobnosti



Rast složenosti raspoređivanja



CNF (Chomsky Normal Form)

Kontekstno slobodna (kontekstno neovisna) gramatika CFG je CNF oblika ako svaki element u P ima sljedeće oblike:

$$A \rightarrow BC \quad A, B, C \in V_N$$

$$A \rightarrow a \quad A \in V_N, a \in V_T$$

Lemma: Za svaku CFG, G postoji njen ekvivalent G' u CNF

ekvivalent $\rightarrow L(G) = L(G')$

Primjeri generiranja nizova za opisivanje uzorka

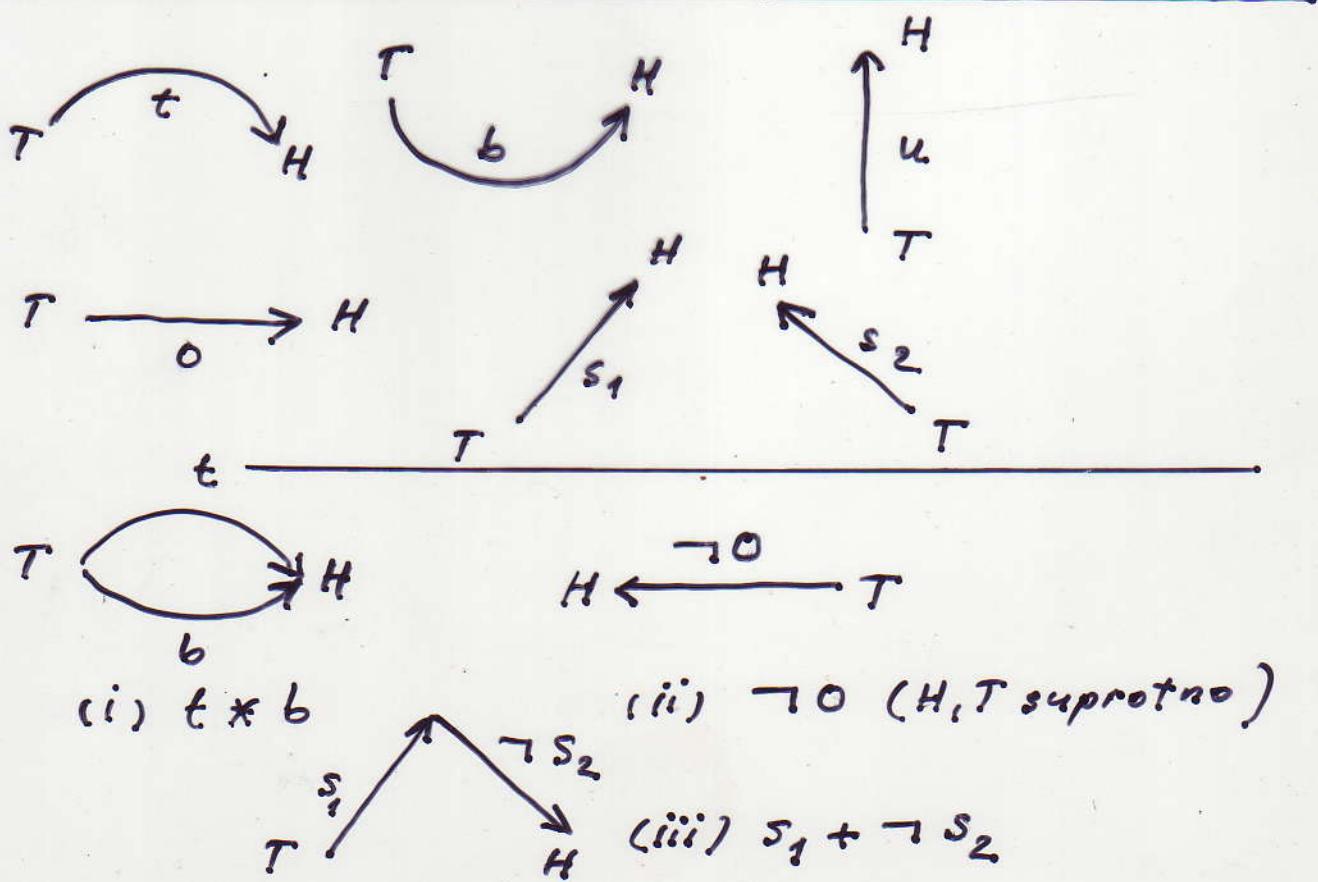
a) 2-D gramatika za opisivanje crteža

- crtež dobiven je digitalne slike nizom postupaka pretprocesiranja (detekcija rubova, granica, linijskih i krivuljnih segmenta)
- crtež → ulaz u sustav za analizu scene koji se temelji na sintaktičnom pristupu

Gramatika za opisivanje rukika:

$$G_{\text{cyk}} = (V_T^{\text{cyk}}, V_N^{\text{cyk}}, P^{\text{cyk}}, S^{\text{cyk}})$$

$$V_T^{\text{cyk}} = \{ t, b, u, o, s, *, \neg, + \}$$

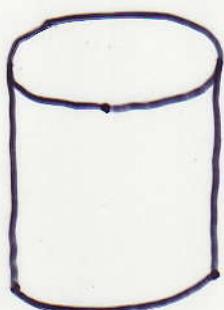


$$V_N^{Cyl} = \{ Top, Body, Cylinder \}$$

P^{Cyl} : Cylinder \rightarrow Top * Body

Top \rightarrow t * b

Body \rightarrow -u + b + u

$$S^{Cyl} = \text{Cylinder}$$


Alternativno:

Cylinder \rightarrow t * b * (-u + b + u)

Uporaba ove vrste gramatike:

(1) Klasifikacija arteža

(2) Odredjivanje strukture entiteta

i eventualno konstenje kao uroda
u neke operacije nad strukturonom
(npr. pomicanje entiteta)

6) Uporaba Picture Description Language (PLD)

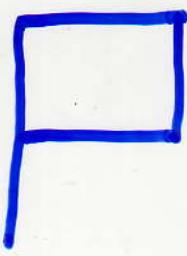
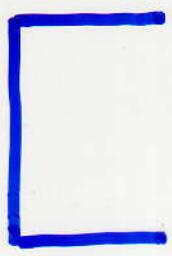
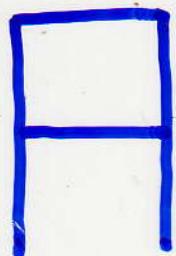
- primitiv u PLD-u n-dimenzionalna
struktura koja ima dve točke:

T (tail; rep) i H (head; glava)



Četiri slovčana ('block') znaka: A, C, P; F

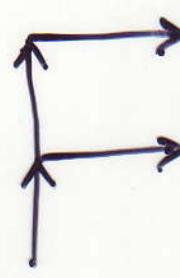
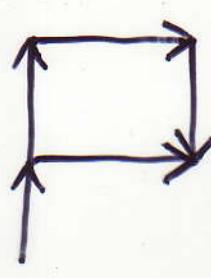
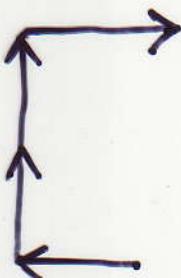
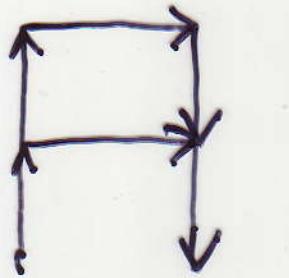
3/4



a) uzorci:

Podskup PLD koristimo za opis znakova!

b) primitivi za prikaz znakova



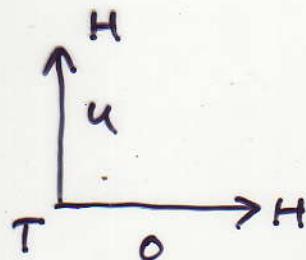
c) Opisi:

$$A = u + ((u \cdot 0 + \neg u) \cdot 0) + \neg u$$

$$C = \neg u + u + u + 0$$

$$P = u + ((u \cdot 0 + \neg u) \cdot 0)$$

$$F = u + (0 \cdot u) + 0$$



(i) $0 \cdot u$ prikaz

c) Kromosomska gramatika (R.S. Ledley, 1964.)

- automatska klasifikacija kromosoma (1964; 1965)

Uporaba kontekstno slobodne gramatike
za klasifikaciju kromosoma u

$C = 2$ razreda

w_1 = submedian

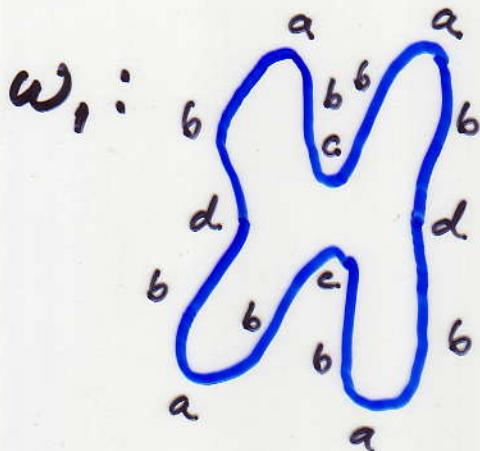
w_2 = telocentric

Zamisao:

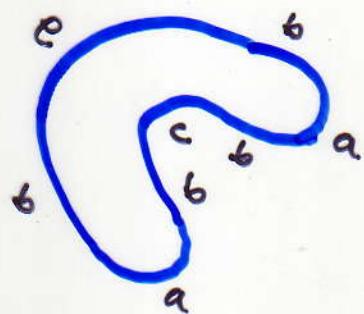
kromosome predaćiti
kao nizove

$\{a, b, c, d, e\}^*$

u skladu s
praviljem
konture
u smjeru
kazaljke
na satu;



$w_2:$



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



$$V_N = \{S, S_1, S_2, A, B, C, D, E, F\}$$

S - početni simbol.

Produktivska pravila P :

$$S \rightarrow S_1$$

$$S_1 \rightarrow AA$$

$$A \rightarrow CA$$

$$A \rightarrow DE$$

$$B \rightarrow bB$$

$$B \rightarrow C$$

$$C \rightarrow Cb$$

$$C \rightarrow d$$

$$D \rightarrow Db$$

$$E \rightarrow cD$$

$$S \rightarrow S_2$$

$$S_2 \rightarrow BA$$

$$A \rightarrow AC$$

$$A \rightarrow FD$$

$$B \rightarrow Bd$$

$$C \rightarrow bC$$

$$C \rightarrow b$$

$$D \rightarrow bD$$

$$D \rightarrow a$$

$$F \rightarrow Dc$$

Zamisao: Ako se koristi prva produkcija u izvođenju rečenice $L(G)$ tj. $S \rightarrow S_1$, tada sekvenca (niz) predstavlja submedian kromosom (w_1)

Ako se radi kao produkcija $S \rightarrow S_2$ tada niz (rečenica) predstavlja telocentric kromosom (w_2)

Znači:

S_1 označava marezd \langle submedian \rangle (w_1)
 S_2 označava marezd \langle telocentric \rangle (w_2)

- A znači \langle arm pair \rangle
- B \langle bottom \rangle
- C \langle side \rangle
- D \langle arm \rangle
- E \langle right part \rangle
- F \langle left part \rangle

Producija $S_1 \rightarrow AA$ odražava činjenicu da je \langle submedian \rangle izgrađen od dva \langle arm pair \rangle

Producija $S_2 \rightarrow BA$ odražava činjenicu da je \langle telocentric \rangle izgrađen od \langle bottom \rangle kojem je pridružen jedan \langle arm pair \rangle

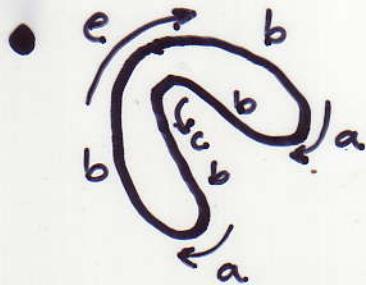
Primjer izvođenja submedian kromosoma:

$$\begin{aligned}
 S \Rightarrow S_1 &\Rightarrow AA \Rightarrow ACA \Rightarrow FDCA \Rightarrow DC DCA \\
 &\Rightarrow bD_c DCA \Rightarrow bDb_c DCA \Rightarrow ba b_c DCA \\
 &\Rightarrow bab_c b DCA \Rightarrow babcb b DCA \Rightarrow babcbab C A \\
 &\Rightarrow babcbab d A \Rightarrow babcbab d A C \Rightarrow babcbab d DEC \\
 &\Rightarrow babcbab d EC \Rightarrow babcbab d ac DC \\
 &\Rightarrow babcbab d aca C \Rightarrow babcbab d aca d
 \end{aligned}$$

/Konishli teor. left most derivaciju!



Primjer izvođenja telocentric kromosoma:

$$\begin{aligned}
 S \Rightarrow S_2 &\Rightarrow BA \Rightarrow eA \Rightarrow eCA \Rightarrow ebA \Rightarrow \\
 &\Rightarrow ebDE \Rightarrow ebD_b E \Rightarrow ebab E \Rightarrow ebabc D \\
 &\Rightarrow ebabc b D \Rightarrow ebabc d D b \Rightarrow ebabc b ab
 \end{aligned}$$


FORMULACIJA PROBLEMA SINTAKTIČKOG RASPOZNAVANJA UZORAKA

- Dva razreda uzoraka $w_1 : w_2$
- Uzori sastavljeni iz komponenti (strukturalnih dijelova) iz konačnog skupa. Te dijelove nazvat ćemo terminalima (primitivima), v_T
- svaki uzorak - niz ili rečenica sastavljena od terminala.
- Pretpostavimo da postoji gramatika G sa svojstvom da se jezik (koji ona generira) sastoji od rečenica (uzoraka) koji isključivo pripadaju samo jednom razredu uzoraka, npr. w_1 .

Takvu gramatiku možemo UPOTREBITI za klasifikaciju budući da zadani uzorak može biti RAZVRETAN u w_1 AKO JE ON (UZORAK) rečenica iz $L(G)$.

U drugom slučaju uzorak je iz w_2 .

Primjer:

Konkretno dobrodružna gramatika: $G = (V_N, V_T, P, S)$ sa $V_N = \{S\}$, $V_T = \{a, b\}$, $P = \{S \rightarrow aaSb, S \rightarrow aab\}$ generira (uzorke) rečenice koje sadrže $2x$ više a-ova od b.

- Formuliramo dva hipotetička razreda uzoraka:

- (1) w_1 uzorci aab, aaaabb, i.t.d
- w_2 uzorci ab, aabb, ... i.t.d

KLASIFIKACIJA = UTVRĐIVANJE DA LI ZADANI NIZ MOŽE BITI GENERIRAN SA GRAMATIKOM G.

Postupak koji se upotrebljava za utvrđivanje da li niz predstavlja rečenicu koja je gramatički korektna za zadani jezik naziva se JEZIČNA ANALIZA (engl. PARSING) ili parsiranje.

Poopcenje problema rpoznavanja na M razreda

- za M razreda , M gramatika,
jezici: $L(G_i)$; $i = 1, 2, \dots, M$

NEPOZNATI UZORAK SE RAZVRSTAVA U RAZRED
 w_i AKO I SAMO AKO JE UZORAK REČENICA
 iz $L(G_i)$.

PROBLEM:

- (1) Kako uzorke najbolje opisati za ovu klasifikacijsku tehniku?
- (2) Kako izabrati gramatike za klasifikaciju?
- (3) Kako riješiti problem utjecaja sume?
- (4) Kako riješiti problem učenja "nintaktičkom pristupu"?

Sintaktično raspoznavanje parsiranjem

- razmatrali generiranje sintaktičnog ili struktturnog opisa složenog uzorka uporabom formalnih gramatičkih

- sada "inverzija" problema: za zadani opis složenog uzorka koji je predovan nizom (ili rečenicom), generiran gramatikom G_i koja odgovara razredu W_i , TREBA ODREDITI KOJEM JEZIKU NIZ Pripada:

$$L(G_i), \quad i=1, 2, \dots, c.$$

intuitivan pristup:

- raspoznavanje podudaranjem nizova
- Imamo G_1, G_2, \dots, G_c gramatika
tjv. class-specific

Zadan je nepoznati uzorak, predovan nizom (rečenicom) x , koji se treba klasificirati.

TREBA ODREDITI DA LI $x \in L(G_i)$ za
 $i=1, 2, \dots, c$

Prepostavka: svaki jezik $L(G_i)$ može se generirati i pohraniti u "biblioteci" (razreduo-zavisne biblioteke)

Podudaranjem niza (rečenice) x sa

svakim pohranjenim uzorkom u svakoj biblioteci ODREDITI Pripadnost x RAZREDU

- Inačica pravila 1-NN (za vektore nizova)

Sličnost dvaju nizova "mjerimo" Levensteinovom udaljenosti

prednosti: - ne zahtijeva jednakih duginu nizova

- manje osjetljivog na manja izobličenja niza
(u odnosu na Hammingovu udaljenost)

Levensteinova udaljenost između nizova

x i y iz V^* predstavlja najmanji broj preslikavanja znakova koji je potreban za pretvorbu niza x u niz y .

Pri tomu su moguća preslikavanja:

1. zamjena znaka:

$$\alpha a \beta \xrightarrow{T_2} \alpha b \beta \quad \text{ta, } b \in V \quad a \neq b \\ \alpha, \beta \in V^*$$

2. brišanje znaka:

$$\alpha a \beta \xrightarrow{T_B} \alpha \beta \quad \text{ta } a \in V \\ \alpha, \beta \in V^*$$

3. umetanje znaka:

$$\alpha \beta \xrightarrow{T_u} \alpha a \beta \quad \text{ta } a \in V \quad \alpha, \beta \in V^*$$

Primjer:

Niz $x = cbabdbbb$ pretvaramo u niz

$y = cbbabbdb$ pomoću najmanje tri preslikavanja:

$$x = cb\cancel{ab}d \xrightarrow{T_2} cb\cancel{a}bbb \xrightarrow{T_2} cbabbdb \xrightarrow{T_2} \\ \xrightarrow{T_2} cb\cancel{bab}bd = y$$

Levenshteinova udaljenost je 3.

Levenshteinova udaljenost:

$$D_L(x, y) = \min_j \left\{ Z_j + B_j + U_j \right\} \quad j=1, 2, \dots, J$$

gdje je

Z_j - broj zamjenjenih znakova

B_j - broj obrisanih znakova

U_j - broj umetnutih znakova

J - broj mogućih pretvorbi
niza x u niz y

- Racunanje Levensteinove udaljenosti dinamickim programiranjem
- Cijena pretvorbe niza x u niz y jednaka je sumi cijena pojedinačnih preslikavanja znakova iz slijeda preslikavanja
- Najmanju cijenu za pretvorbu možemo izracunati DINAMICKIM PROGRAMIRANJEM tj. sljedujim minimizirajućem djelomičnih suma.

Problemi kod pristupa podudaranjem nizova:

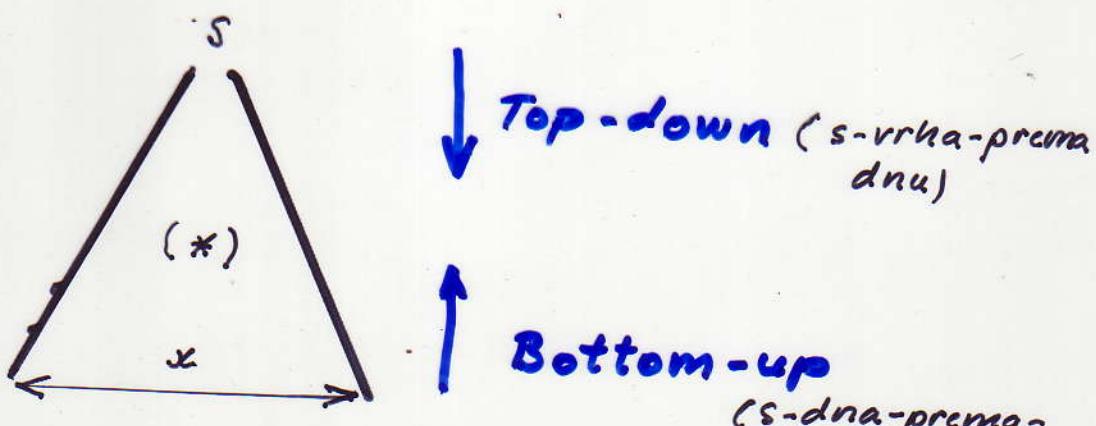
1. često je $|L(G_i)| = \infty$ zato je potraua nizova u "biblioteku" za razred W_i nemoguća
2. čak i kada je $L(G_i)$ prebrojivo "biblioteke" su vrlo velike
3. Računski zahtjevna procedura podudaranja

Parsiranje (jezična analiza)

- zadan je niz terminala (recenica) x
- zadana je gramatika G :

$$G = (V_T, V_N, P, S)$$

- Trebamo "trogut":



- Postupak "popunjavanja" unutrašnjosti trokuta produkcijama koje povezuju S sa x naziva se parsiranje.
- Ako se uspije u tome vrijedci da je $x \in L(G)$

Cocke-Younger-Kasami (CYK)

algoritam parsiranja

CYK algoritam zahtijeva da kontekstno slobodna gramatika CFG bude u Chomsky Normal Form (CNF).

Za CFG u CNF dopuštene su produkcije tipa

$$A \rightarrow BC$$

$$A \rightarrow a$$

/Nije dopuštena produkcija $A \rightarrow \epsilon/$

- Zadan je niz $x = a_1, a_2, \dots, a_n$ koji se treba parsirati
- konstruiraj trokutastu tablicu s elementima t_{ij} , $1 \leq i \leq n$, $1 \leq j \leq (n-i+1)$, gdje indeksi i i j odgovaraju stupcima, odnosno recima. (Ishodišni element ($i=1, j=1$) smješten je u donjem lijevom kutu).
- Svaki element tablice t_{ij} mora biti konstruiran tako da sadrži podskup od V_N s neterminalom A unesenim u t_{ij} ako $\overset{se}{V}$ podniz od x / započinje s a_i i širi se za j simbola (znakova) / može izvesti iz A :

$$A \underset{G}{\Rightarrow} a_i \dots a_{i+j-1}$$

- x je u $L(G)$ ako i samo ako je s u t_{in} kada se postupak oblikovanja (punjenja) tablice završi.

Opaska: Tablica se gradi s lijeva udesno, započevši od najnižeg redka.

Algoritam:

1. Korak: Postavi $j=1$. Izračunaj t_{i1} , $1 \leq i \leq n$ smještanjem A u t_{ij} u skladu s produkcijskom $A \rightarrow a_i$ iz P .

2. Korak: Pretpostavljajući da su elementi tablice t_{ij} , $i=1 \dots n$, već određeni za $1 \leq j \leq n$, izračunaj t_{ij} tako da se A smještava u t_{ij} kada za neki k takav da $j \leq k \leq j$ postoji produkcijska rečenica $A \rightarrow BC$ u P sa B u t_{ik} i C u $t_{i+k, j-k}$.

Ovo je rekursivni korak koji se temelji na dekompoziciji podniza $a_i \dots a_{i+j-1}$ na prefiks $a_i \dots a_{i+k-1}$ i

sufiks podniza $a_{i+k} \dots a_{j-k}$ tako da vrijedi

$$B \xrightarrow{*} a_i \dots a_{i+k-1} \quad i$$

$$C \xrightarrow{*} a_{i+k} \dots a_{j-k} \quad i \quad A \xrightarrow{*} BC /$$

3. Korak: Ponavlja se dok se ne izgradi tablica ili dok cijeli redak nema \emptyset .

x je iz $L(G)$ ako i samo ako je S u t_{1n} .

Primjer:

Zadana je gramatika sa sljedećim produkcijama:

$$S \rightarrow AB / BB$$

$$A \rightarrow CC / AB / a$$

$$B \rightarrow BB / CA / b$$

$$C \rightarrow BA / AA / b$$

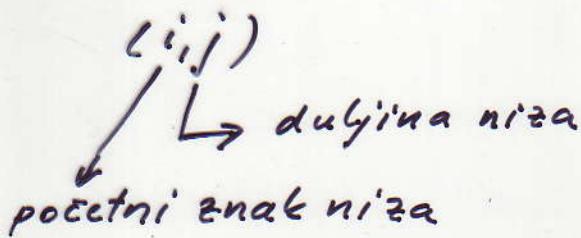
Zadan je niz (recenica) $x = aabb ; |x| = 4$

$$x \in L(G) ?$$

Konstrukcija trokutaste tablice: t_{ij}

j				
4	t_{14}			
3	t_{13}	t_{23}		
2	t_{12}	t_{22}	t_{32}	
1	t_{11}	t_{21}	t_{31}	t_{41}
	1	2	3	4
$x =$	a	a	b	b
				i

element u tablici



1. Korak: $j=1$

Element u (i, j) odgovara mogućnosti produkcije niza duljine j koji započinje znakom a_i / Elementi ($1, 1$), ($2, 1$), ($3, 1$) i ($4, 1$) su slijedno: A, A, B, B.

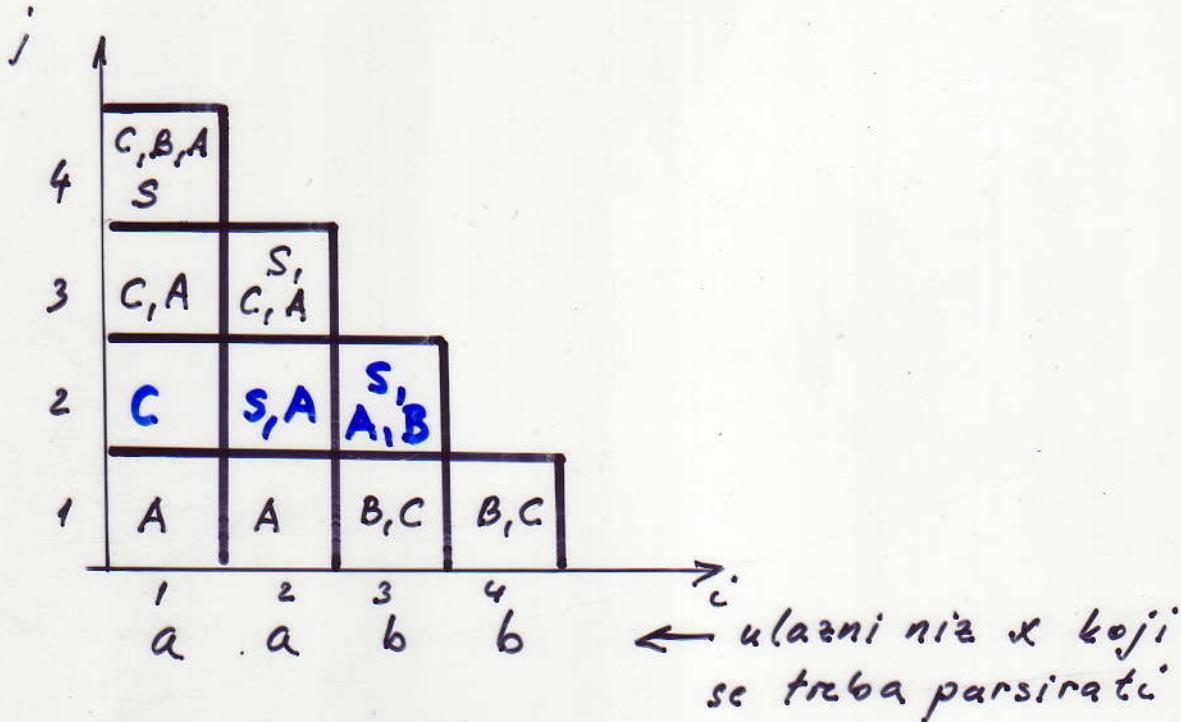
$j=1 \quad t_{1j}, 1 \leq i \leq 4$

za $a_1 = a$ imamo $t_{11} = \{A\}$ zato što je produkcija $A \rightarrow a$ u P

za $a_2 = a \quad t_{21} = \{A\}$

za $a_3 = b \quad t_{31} = \{B, C\}$, zato što je $B \rightarrow b$ u P i $C \rightarrow b$

za $a_4 = b \quad t_{41} = \{B, C\}$



2. Korak:

t_{12} podniz zapocinje s $a_1 = a$ i duljine je 2

aa je rezultat produkcije koja $(a)(a)$ na desnoj strani ima AA

u P je to produkcija

$$C \rightarrow AA$$

$$C \in (1, 2)$$

t_{22} podniz započinje s a i duljina je 2
ab

(a)(b) podniz je rezultat produkcija
koje imaju na desnoj strani

AB i AC

to su produkcije

$$S \rightarrow AB$$

$$A \rightarrow AB$$

$$S, A \in (2, 2)$$

ispitati

t_{21}

t_{31}

$$t_{21}, \dots, A \quad t_{31}, \dots, B, C$$

t_{32} podniz započinje s b i duljina je 2

bb

(b)(b) podniz je rezultat produkcija
koje na desnoj strani imaju

BB

CC

BC

CB

iz P su to:

t_{31}, \dots, B, C

$$S \rightarrow BB \checkmark$$

t_{41}, \dots, B, C

$$A \rightarrow CC \checkmark$$

$$B \rightarrow BB \checkmark$$

$j=3$

t_{13} odgovara nizu duljine 3 koji započinje s terminalom $a_1 = a$

aab

$(a)(ab)$

$(aa)(b)$

$(a)(ab)$ treba ispitati c'eliju $(1,1) \dots A$

treba ispitati c'eliju $(2,2) \dots S, A$

imamo desnu stranu
produkcija

AS

AA

iz P dobivamo

$C \rightarrow AA$

$(aa)(b)$ treba ispitati c'eliju $(1,2) \dots C$

treba ispitati c'eliju

t_{31}

$(3,1) \dots B, C$

imamo desnu stranu
produkcija

CB

CC

iz P dobivamo:

$A \rightarrow CC$

$C, A \in (1,3)$

t_{23}

podniz je ab6

(a)(66) treba ispitati

(2,1) ... A

treba ispitati (3,2) ... S, A, B

- desna strana produkcija

AS

AA

^{AB}

iz P dobivamo:

$C \rightarrow AA$

$A \rightarrow AB$

(ab)(6) treba ispitati (2,2) ... S, A

i (4,1) ... B, C

- desna strana
produkcija

SB

SC

AB

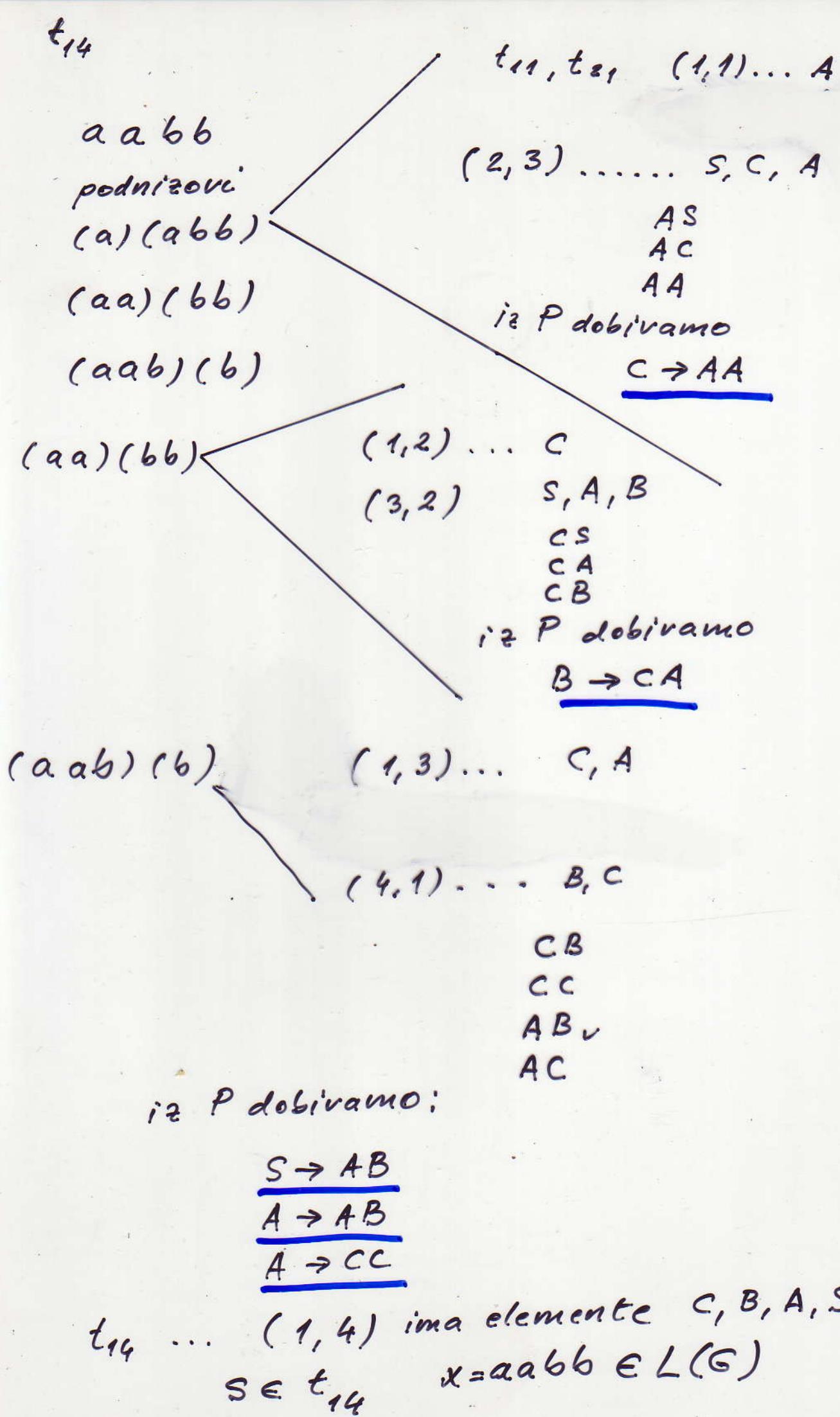
AC

iz P dobivamo:

$S \rightarrow AB$

$A \rightarrow AB$

$S, C, A \in (2,3)$



Primer:

$$G = (\{S, A, B, C\}, \{a, b\}, P, S)$$

$$P: \quad S \rightarrow AB$$

$$S \rightarrow AC$$

$$A \rightarrow a$$

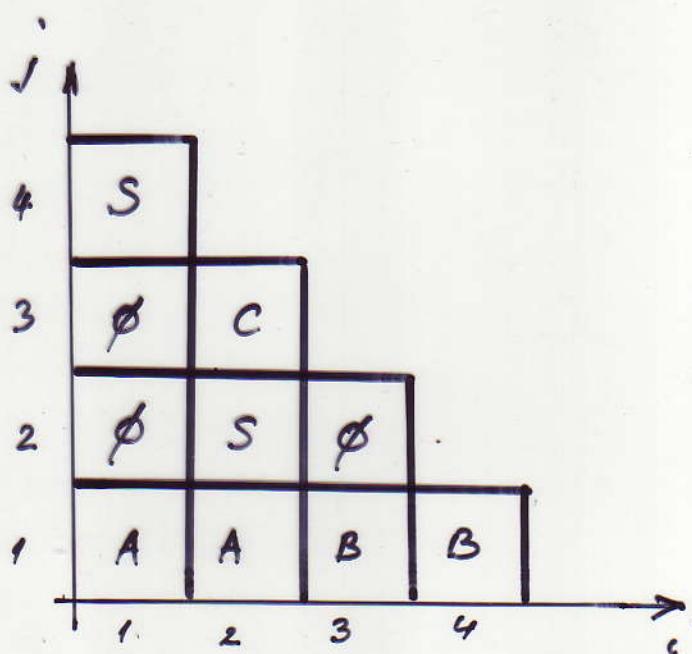
$$B \rightarrow b$$

$$C \rightarrow SB$$

generira jezik

$$x \in L(G) = \{x | x = a^m b^m, m \geq 1\}$$

Parsiranje $x = aaabbb$



$$x = aaabbb \in L(G)$$

Primjer: zadana je regularna gramatika
(FSG) s produkcijama

$$S \rightarrow A_3 A_1$$

$$A_3 \rightarrow b$$

$$S \rightarrow A_4 A_2$$

$$A_4 \rightarrow c$$

$$A_1 \rightarrow A_3 A_2$$

$$A_2 \rightarrow b$$

$$A_2 \rightarrow A_5 A_2$$

$$A_5 \rightarrow a$$

Parsirati nizove

$$x = bbaab$$

i

$$x = cbaab$$

utvrditi pripadnost $L(G)$

Rješenje:

<i>i</i>	5	S			
4	-	A_1			
3	-	-	A_2		
2	A_1	\emptyset	\emptyset	A_2	
1	$A_2 A_3$	$A_2 A_3$	A_5	A_5	$A_2 A_3$
	b	b	a	a	b

$x = bbaab \in L(G)$

Stohastičke gramatike

- Do sada smo pretpostavljali da vrijedi:

$$(i) \quad L(G_i) \cap L(G_j) = \emptyset \quad i$$

(ii) Greške u generiranju rečenica
uporabom gramatike se ne događaju

- Nadalje, nismo imali načina ugraditi
apriornu informaciju koja se odnosi na
vjerojatnost razreda uzorka

$P(G_i)$ - mjera vjerojatnosti gramatike
 G_i

Stohastička gramatika:

$$G_s = \{V_N, V_T, P_s, S_s\}$$

P_s i S_s

↳ skup stohastičkih produkcija oblika:

$$\alpha_i \xrightarrow{P_{ij}} \beta_j, \quad (P_{ij}: \alpha_i \rightarrow \beta_j)$$

gdje je P_{ij} vjerojatnost da će α_i
zamijeniti β_j .

S_s je modifikacija S , gdje je rečenica

$S \in V_N$ opisan vjerojatnosnom distribucijom.

Generiranje recenica iz stohastičkog jezika

- pretpostavimo da je niz x generiran uporabom n produkcija:

$$S_s \Rightarrow x_0 \Rightarrow x_1 \Rightarrow x_2 \dots \Rightarrow x_n = x$$

- Označimo i -tu produkciju iz P_s kao p^i ; $i=1, 2, \dots, m$ i neka bude $t_{k-1, k}$ oznaka ili labela produkcije koja je korištena da se x_{k-1} napiše kao x_k , $k=1, 2, \dots, n$

$$\text{Tako imamo } t_{k-1, k} = p^i$$

Cijela sekvenca produkcija upotrebljena za generiranje x ima odgovarajuću vjerojatnost.

Vjerojatnost generiranja x uporabom sekvence stohastičkih produkcija $t_{0,1}, t_{1,2}, \dots, t_{n-1,n}$ dana je s vjerojatnošću

$$P(t_{0,1} \cap t_{1,2} \cap \dots \cap t_{n-1,n})$$

$$P(t_{0,1} \cap t_{1,2} \cap \dots \cap t_{n-1,n}) =$$

$$P(t_{n-1,n} | t_{n-2,n-1}, \dots, t_{1,2}, t_{0,1}) \cdot P(t_{n-2,n-1} |$$

$$t_{n-3,n-2}, \dots, t_{1,2}, t_{0,1}) \dots$$

$$P(t_{1,2} | t_{0,1}) \cdot P(t_{0,1})$$

536

- Ako se ujetne vjerojatnosti zamijene s bezvjetnim vjerojatnostima:

$$P(t_{k-1,k} | t_{k-2,k-1}, \dots) = P(t_{k-1,k})$$

dobiva se neograničena stohastička gramatika
/ To odgovara pretpostavci da za generiranje
niza x , vjerojatnost primjene sljedeće
produkuje nezavisno od prethodno
izabrane sekvence /

$$P(t_{0,1}, t_{1,2}, \dots t_{n-1,n}) = \prod_{q=1}^n P(t_{q-1,q})$$

stohastički jezik generiran gramatikom
 G_s sastoji se od svih izvodljivih
nizova i njima pridruženim vjerojatnostima

$$L(G_s) = \{ (x, p(x)) \mid x \in V_T^*, S_s \xrightarrow{P_s} x, \\ j = 1, 2, \dots, k \text{ i } p(x) = \sum_{j=1}^k p_j \}$$

Primjer

$$G_s = \{V_T, V_N, P_s, S_s\}$$

$$V_T = \{a, b\}, V_N = \{S_s, A, B\} \text{ i }$$

$$\begin{aligned} P_s : \quad S_s &\xrightarrow{1} bA \\ &A \xrightarrow{0.8} aB \qquad P(S_s) = 1 \\ &A \xrightarrow{0.2} b \\ &B \xrightarrow{0.3} a \\ &B \xrightarrow{0.7} bS_s \end{aligned}$$

Primjenom produkcijskih pravila generiraju se nizovi (rečenice) jezika $L(G_s)$:

$$\begin{aligned} &b, \quad ba, \quad (bab)^n b \text{ i} \\ &(bab)^n ba \end{aligned}$$

rečenice bb dobivamo: (1. pravilo + 3. pravilo)

$$S_s \xrightarrow{1} bA \xrightarrow{0.2} bb$$

-vjerojatnost rečenice bb , je

$$p(bb) = 1 \cdot 0.2 = 0.2$$

rečenice baa dobivamo: (1. + 2. + 4.)

$$S_s \xrightarrow{1} bA \xrightarrow{0.8} baB \xrightarrow{0.3} baa$$

$$p(baa) = 1 \cdot 0.8 \cdot 0.3 = 0.24$$

rečenice $(bab)^n b$ dobivamo ponavljajući prvo, drugo i peto produkcijsko pravilo te sa primijenom pravilom:

- Vjerojatnost rečenice $(bab)^n b b$ za $n \geq 1$ je :

$$p((bab)^n b b) = (0.56)^n \cdot 0.2$$

- Rečenice $(bab)^n b a a$ dobivamo ponavljajenjem pravog, drugog i petog produkcijskog pravila te primjenom drugog i četvrtog pravila

- Vjerojatnost rečenica $(bab)^n b a a$ za $n \geq 1$ je :

$$p((bab)^n b a a) = (0.56)^n \cdot 0.24$$

Za svaku stohastičku gramatiku vrijedi:

$$\sum_{x \in L(G_s)} p(x) = 1$$