نظریه زبانها دکتر کارگهی فصل ۶ و ۷

①

Simplification of Context-free grammars

21 November 2010
19:28

ساده سازی گرامرهای مستقل از متن و فرم های نرمال

اگر این دو قاعده را داشتیم

$$\left.\begin{array}{l} A \to x_1 B x_2 \\ B \to y_1 | \cdots | y_n \end{array}\right\} \Rightarrow A \to x_1 y_1 x_2 | \cdots | x_1 y_n x_2$$

بجای B خط پایین را گذاشته

اگر این سه را داشته باشد

۱- λ نباید شد

۲- unit ،

۳- قواعد بدون استفاده

ساده شده

Removing Useless Productions حذف قواعد بی فایده

$S \to aSb | \lambda | A$

$A \to aA$ → چون به رشته نمی رسیم حذف می کنیم (از A خارج می شویم).

تعریف :

A : useful iff $\exists w \in L(G) :: S \overset{*}{\Rightarrow} xAy \overset{*}{\Rightarrow} w$

otherwise A : useless

تعریف : به غیرپایانه A مفید (useful) گفته می شود اگر و تنها اگر حداقل یک رشته w متعلق به L(G) وجود داشته باشد که به ازای آن داشته باشیم $S \overset{*}{\Rightarrow} xAy \overset{*}{\Rightarrow} w$

یعنی از S بتوانیم برسیم به A و در ضمن به رشته برسیم.

غیرپایانه A مفید است اگر و تنها اگر در مراحل یک اشتقاق ظاهر شده باشد.

در غیر اینصورت به غیرپایانه A بی فایده گفته می شود.

مثال) $S \to A$
$A \to aA | \lambda$
$B \to bA$ → useless بی فایده

مثال)
$S \to aS | A | B$
$A \to a$
$B \to aa$ → (بی فایده) B چون بهش نمی رسیم
$C \to aCb$ → (بی فایده) C چون از بین می ره

Removing λ-productions

تعریف :

$A \to \lambda$: λ-production → λ

$A \overset{*}{\Rightarrow} \lambda$: nullable (پوچ) → از A طی اشتقاق چند مرحله به λ برسیم

مثال) $S \to aS_1b$
$S_1 \to aS_1b | \lambda$ $\xrightarrow{nullable=\{S_1\}}$ $S \to aS_1b | ab$
$S_1 \to aS_1b | ab$

amitiisweb.com

$S_1 \to aS_1b \mid \lambda \longrightarrow S_1 \to aS_1b \mid ab$

مثال)
$S \to ABaC$
$A \to BC$
$B \to b \mid \lambda$
$C \to D \mid \lambda$
$D \to d$

nullable = {B, C, A}

λ به A | λ به B | λ به d | ab به ab

$S \to ABaC \mid BaC \mid AaC \mid ABa \mid aC \mid Ba \mid Aa \mid a$

C به هر

$A \to BC \mid C \mid B$
$B \to b$
$C \to D$
$D \to d$

معادل قبلی اما قواعد λ تولید وجود ندارد.

* از هر گرامری که زبان آن فاقد رشته λ باشد، می‌توان بدون تغییر زبان قواعد λ را حذف کرد.

Removing Unit-Productions

حذف قواعد Unit (واحد یا یکه)

تعریف :

$A \to B$: Unit-Production

مثال)
$S \to Aa \mid B$
$B \to A \mid bb$
$A \to a \mid bc \mid B$

$S \to Aa$
$B \to bb$
$A \to a \mid bc$

$S \to bb \mid a \mid bc$
$B \to a \mid bc$
$A \to bb$

$S \to \cancel{B} \to A \mid bb$

$A \to a \mid bc \mid B$

$S \to Aa \mid bb \mid a \mid bc$
$B \to bb \mid a \mid bc$
$A \to a \mid bc \mid bb$

مراحل ساده سازی یک گرامر :

1. حذف قواعد λ

amirisweb.com

(۲) حذف قواعد Unit
(۳) حذف قواعد Useless

مثال)

$S \to AaB \mid aaB$
$A \to \lambda$
$B \to bbA \mid \lambda$

$\xrightarrow{\text{حذف } \lambda}$ nulloble $= \{A, B\}$

$S \to AaB \mid aB \mid Aa \mid a \mid aaB \mid aa$
X
$B \to bbA \mid bb$

A وجود ندارد

$\xrightarrow{\text{حذف Unit}}$ نداریم

$\xrightarrow{\text{حذف useless}}$

$S \to aB \mid a \mid aaB \mid aa$
$B \to bb$

amitisweb.com

فرم‌های نرمال (Normal Forms)

CNF : Chomskey Normal Form

$A \to BC$
$A \to a$

$A, B, C \in V$
$a \in T$

تعدادی در این زبان گرامر نداریم.

مثال)

$S \to ABa$
$A \to aab$
$B \to Ac$

$\longrightarrow$

$S \to ABX_a$
$X_a \to a$
$A \to X_aX_aX_b$
$X_b \to b$
$B \to AX_c$
$X_c \to c$

$\longrightarrow$

$S \to AZ_1$
$Z_1 \to BX_a$
$X_a \to a$
$A \to X_aZ_2$
$Z_2 \to X_aX_b$
$X_b \to b$
$B \to AX_c$
$X_c \to c$

تبدیل شد به CNF

آیا هر زبان مستقل از متن فرم CNF دارد؟

* هر زبان مستقل از متن که ناتهی و λ باشد دارای گرامر به فرم نرمال چامسکی است.

مراحل تبدیل:

۱- حذف قواعد λ
۲- حذف قواعد Unit
۳- دنبال کردن روند مشابه مثل قبل.

$A \to aab$ $\longrightarrow$ $A \to X_a Z_1$

$B \to abc$ $\longrightarrow$ $B \to Z_1 X_c$

* تعداد مراحل لازم برای تولید رشته‌ای بطول n چند است؟ (مهم مهم با حالت بوده)

$S \overset{n-1}{\Rightarrow} A_1 \cdots A_n \overset{n}{\Rightarrow} a_1 \cdots a_n$

در $n-1$ مرحله می‌رسیم به n ، نان ترمینال .

$n-1+n=2n-1$

GNF: Greibach Normal Form

$\to A \to aX$

$\begin{cases} a \in T \\ X \in V^* \end{cases}$

مثال) $S \to AB$
$A \to aA \mid bB \mid b$
$B \to b$

$\longrightarrow$ مشکل این است که باید مقادیر A, B را از خط پایین جایگزین کرد.

$S \to aAB \mid bBB \mid bB$

$\vdots$

مثال) $S \to abSb \mid aa$

$S \to aX_bSX_b \mid aX_a$
$X_a \to a$
$X_b \to b$

* اگر یک زبان مستقل از متن فاقد رشته λ باشد ، برای آن گرامر GNF وجود دارد.

* مراحل تبدیل : ۱- حذف چپ‌گردی ، ۲- حذف قواعد λ ، ۳- حذف قواعد Unit ، ۴- جایگزاری

* حذف چپ‌گردی :

$A \to A\alpha_1 \mid \cdots \mid A\alpha_n$
$A \to \beta_1 \mid \cdots \mid \beta_m$

$(\beta_1 + \cdots + \beta_m)(\alpha_1 + \cdots + \alpha_n)^*$

$A \to \beta_1 A' \mid \cdots \mid \beta_m A'$
$A' \to \alpha_1 A' \mid \cdots \mid \alpha_n A' \mid \lambda$

amitisweb.com

5

* $(\beta_1 + \cdots + \beta_m)(\alpha_1 + \cdots + \alpha_n)$

$A \to \beta_1 A' \mid \cdots \mid \beta_m A'$
$A' \to \alpha_1 A' \mid \cdots \mid \alpha_n A' \mid \lambda$

* تعداد مراحل مورد نیاز برای تولید رشته ای بطول n در هر مرحله زیر در هر مرحله Tran

یک الگوریتم عضویت برای گرامرهای مستقل از متن (CYK)

Created with Microsoft Office OneNote 2007
One place for all your notes and information

الگوریتم عضویت CYK

در کتاب بخوانید.

amitisweb.com

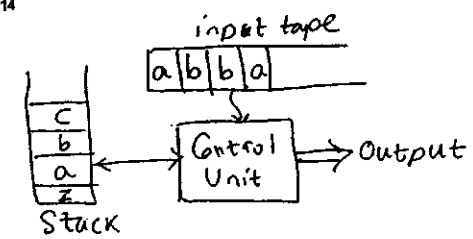
بخش 6_1 (6-13, 20-21-22 و 23-25)

بخش 6_2 4-6-8-9, 14, 15, 16

از آزمون حذف شده.

Down Automata (PDA)

28Sunday, November
05:14



PDA → NPDA
PDA → DPDA

NPDA:

$M: (Q, \Sigma, \Gamma, \delta, q_0, z, F)$

مجموعه حالات

الفبای زبان

الفبای Stack

Stack Start symbol

$z \in \Gamma$

$\delta: Q \times (\Sigma \cup \{\lambda\}) \times \Gamma \to$ finite subsets of $Q \times \Gamma^*$

توضیح انواع حرکات قابل انجام توسط یک PDA:

تبدیلات

$\delta(q_0, a, c) = \{(q_1, xc), (q_1', xxc)\}$

هر حرکتی بخواهیم کنیم حتماً باید یک حرکت انجام دهد.

$\delta(q_1, b, x) = \{(q_2, y)\}$

$\delta(q_2, b, y) = \{(q_3, \lambda)\}$

$\delta(q_3, \lambda, c) = \{(q_4, c)\}$

یعنی بدون توجه به ورودی

Instantaneous Description (توصیف لحظه‌ای)

(q_i, α, β)

حالت جاری

باقی‌مانده رشته ورودی

محتوای Stack

چون عدم قطعیت داشتیم این حرکت

Satisfy

$(q_0, abba, cbaz) \vdash (q_1, bba, xcbaz)$

$\vdash (q_1', bba, xxcbaz)$

amitisweb.com

Stack خالی باشد یعنی ماشین دیگر نمی‌تواند حرکت کند ⇐ یعنی ماشین از کار افتاده

⑦

$\vdash (q_1', bba, xxcbaz)$

پذیرش توسط یک PDA :

۲ روش پذیرش :

1- Acceptance by final state $(M:(Q,\Sigma,\Gamma,\delta,q_0,z,F))$

$L(M)=\{w\in\Sigma^* \mid (q_0,w,z)\vdash^* (q_f,\lambda,\gamma),\ q_f\in F,\ \gamma\in\Gamma^*\}$

پذیرش Stack خالی

2- Acceptance by empty stack $(M:(Q,\Sigma,\Gamma,\delta,q_0,z))$

$L(M)=\{w\in\Sigma^* \mid (q_0,w,z)\vdash^* (p,\lambda,\lambda),\ p\in Q\}$

stack خالی شود حتی z را هم ندارد.

مثال :

$L=\{a^nb^n \mid n\geq 0\}$

$\delta(q_0,\lambda,z)=\{(q_f,z)\}$

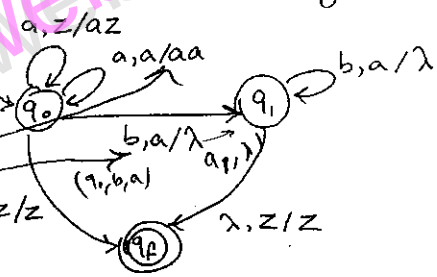
$\delta(q_0,a,z)=\{(q_0,az)\}$

$\delta(q_0,a,a)=\{(q_0,aa)\}$

$\delta(q_0,b,a)=\{(q_1,\lambda)\}$

$\delta(q_1,b,a)=\{(q_1,\lambda)\}$

$\delta(q_1,\lambda,z)=\{(q_f,z)\}$



مثال a^3b^3

$(q_0,aaabbb,z)\vdash(q_0,aabbb,az)\vdash(q_0,abbb,aaz)$

$\vdash(q_0,bbb,aaaz)\vdash(q_1,bb,aaz)\vdash(q_1,b,az)$

$\vdash(q_1,\lambda,z)\vdash(q_f,\lambda,z)$

به پایان رشته باشیم پذیرفته شده

که در اینجا a^3b^3 پذیرفته شده است

چون پایان رشته‌ایم.

$L=\{a^nb^{n+1} \mid n\geq 0\}$

$n=0$ را ← $\delta(q_0,b,z)=\{(q_f,z)\}$

$\delta(q_0,a,z)=\{(q_0,az)\}$

$\delta(q_0,a,a)=\{(q_0,aa)\}$

$\delta(q_0,b,a)=\{(q_1,\lambda)\}$

$\delta(q_1,b,a)=\{(q_1,\lambda)\}$

$\delta(q_1,\lambda,z)=\{(q_f,z)\}$

$L=\{ww^R \mid w\in\{a,b\}^*\}$

برای رشته λ $\delta(q_0,\lambda,z)=\{(q_f,z)\}$

amtisweb.com

⑧ ww^R را تعریف کنیم

جزء زبان‌هایی است که پیاده‌ساز غیرقطعی دارد زیرا وسط رشته را باید بدانیم کجا است که

$\delta(q_0,a,z)=\{(q_0,az)\}$
$\delta(q_0,b,z)=\{(q_0,bz)\}$
$\delta(q_0,a,a)=\{(q_0,aa),(q_1,\lambda)\}$
$\delta(q_0,b,a)=\{(q_0,ba)\}$
$\delta(q_0,a,b)=\{(q_0,ab)\}$
$\delta(q_0,b,b)=\{(q_0,bb),(q_1,\lambda)\}$

$\delta(q_1,a,a)=\{(q_1,\lambda)\}$
$\delta(q_1,b,b)=\{(q_1,\lambda)\}$
$\delta(q_1,\lambda,z)=\{(q_f,z)\}$

push و pop کنیم.

طبق غیرقطعی فرض کنیم دوتا aa دوتا bb وسط رشته است.

$L=\{a^nb^{2n}\mid n\geq 0\}$

به ازای هر a یک علامت بگذاریم 2b برداریم.

$L=\{w\in\{a,b\}^*\mid n_a(w)=n_b(w)\}$ مجموع دو کلمه

برای هر a، push کرده یا هر b یک a را پوپ کنیم.
بالای stack نگاه کرده

$L=\{a^nb^mc^{n+m}\mid n,m\geq 0\}$

به ازای هر a یک علامت گذاشته به ازای هر b یک علامت گذاره

$\delta(q_0,b,z)=\{(q_1,xz)\}$
$\delta(q_0,a,z)=\{(q_0,xz)\}$
$\delta(q_0,a,x)=\{(q_0,xx)\}$
$\delta(q_0,b,x)=\{(q_1,xx)\}$
$\delta(q_1,b,x)=\{(q_1,xx)\}$
$\delta(q_1,c,x)=\{(q_2,\lambda)\}$

هرگاه a دیدیم
هرگاه b دیدیم
می‌رویم q_1 تا دیگر a نقش نداره.

الفبای stack: xz

$\delta(q_0,b,z)=\{(q_0,bz)\}$
$\delta(q_0,a,z)=\{(q_0,az)\}$
$\delta(q_0,a,a)=\{(q_0,aa)\}$
$\delta(q_0,b,a)=\{(q_0,ba)\}$
$\delta(q_0,b,b)=\{(q_0,bb)\}$
$\delta(q_0,c,b)=\{(q_1,\lambda)\}$
$\delta(q_0,c,a)=\{(q_1,\lambda)\}$

وقتی c دیدیم یک pop کنیم
تا خالی شود stack
تغییر الفبا

الفبای stack: a,b,z

می‌توان الفبای stack را کوچک داد.
تعداد state، stack را بالا داد.

$L=\{a^3b^nc^n\mid n\geq 1\}$

$\delta(q_0,a,z)=\{(q_1,z)\}$
$\delta(q_1,a,z)=\{(q_2,z)\}$
$\delta(q_2,a,z)=\{(q_3,z)\}$
$\delta(q_3,b,z)=\{(q_4,bz)\}$

توی stack نمی‌ذاریم
چون به درد ما نمی‌خوره
تعداد q_4 می‌رویم چون یکی
شیم یک b داریم.

$L=\{a^{3n}b^nc^n\mid n\geq 1\}$

$\delta(q_0,a,z)=\{(q_1,az)\}$

amitisweb.com

راه حل بهتر: هر کاراکتر را ۱ بار بریم

$\delta(q_{14},a,a)=\{(q_{15},aa)\}$
$\delta(q_{15},a,a)=\{(q_{16},\lambda)\}$
$\delta(q_{16},a,a)=\{(q_{16},\lambda)\}$
$\delta(q_{16},b,z)=\{\ldots$

$\delta(q_0,a,z)=\{(q_1,\overset{29}{az})\}$
$\delta(q_1,a,a)=\{(q_1,\lambda)\}$

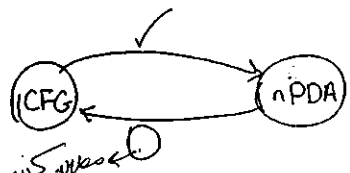
و 29 ...

سمبل الصلاح گرفته

$\delta(q_0,a,z)=\{(q_0,1z)\}$
$\delta(q_0,a,1)=\{(q_0,2)\}$
$\vdots$
$\delta(q_0,a,\overset{29}{z})=\{(q_0,z')\}$
$\delta(q_0,b,z')=\ldots$

The relation between CFG and NPDAs

رابطه بین CFG و NPDA



در کتاب است

مطالعه کنید

مستقل از متن

CFG → nPDA

فرض می‌کنیم که فرم نرمال است.

$$\begin{cases} S\to aA \\ A\to aABC \mid bB \mid a \\ B\to b \\ C\to c \end{cases}$$

چون شروع از حالت آغاز q_0.

همیشه ثابت هستند
$$\begin{cases} \delta(q_0,\lambda,z)=\{(q,Sz)\} \\ \delta(q,\lambda,z)=\{(q_f,z)\} \end{cases}$$

چون قانون اول به S نیاز دارد در اول Stack

چون نمی‌تواند به حالت q_f

برای ورودی به اول توجه به ورودی

$S\to aA$
$\delta(q,a,S)=\{(q,A)\}$

$A\to aABC$
$\delta(q,a,A)=\{(q,ABC)\}$

$A\to bB$
$\delta(q,b,A)=\{(q,B)\}$

$A\to a$
$\delta(q,a,A)=\{(q,\lambda)\}$

$B\to b$
$\delta(q,b,B)=\{(q,\lambda)\}$

$C\to c$
$\delta(q,c,C)=\{(q,\lambda)\}$ خروجی است

مثال تولید این رشته aaabc

$$S\Rightarrow aA\Rightarrow aaABC\Rightarrow aaaBC\Rightarrow aaabC\Rightarrow aaabc$$

$$(q_0,aaabc,z)\vdash(q,aaabc,Sz)\vdash(q,aabc,Az)\vdash(q,abc,ABCz)\vdash(q,bc,BCz)$$
$$\vdash(q,c,Cz)\vdash(q,\lambda,z)\vdash(q_f,\lambda,z)$$

* نتیجه: به ازای هر زبان مستقل از متن یک nPDA با سه حالت وجود دارد.

q_0 q q_f

پس فقط ۳ state است.

آیا می‌توان این را به ۲ state تغییر داد؟ تمرین کتاب.

amitisweb.com

بخش ۲-۷ : 5-6-8-9-15

بخش ۳-۷ : 2-4-6-7-8

(12 تا 14) یکیش حل کن

15-18 همه مهم

(20)

چه موقع یک PDA ، یک DPDA محسوب می‌شود ؟

باید دو شرط زیر برقرار باشد :

۱. $\delta(q,a,b)$ حداکثر یک عضو داشته باشد. (یک مجموعه) که این مجموعه یک عضو داشته باشد.

۲. اگر $\delta(q,\lambda,b)$ خالی نیست، آنگاه $\delta(q,c,b)$ به ازای تمام $c \in \Sigma$ خالی باشد.

$\delta(q,\lambda,b)$ خالی نیست

$\delta(q,c,b)$ خالی باشد

بدون توجه به ورودی (b) ، یا مد پس باید با توجه به ورودی c باید خالی باشد.

یک DPDA برای زبان $\{a^n b^n \mid n \geq 0\}$

$M = (\{q_0, q_1, q_2\}, \{a,b\}, \{0,1\}, \delta, q_0, 0, \{q_0\})$

$\delta(q_0, a, 0) = \{(q_1, 10)\}$
$\delta(q_1, a, 1) = \{(q_1, 11)\}$
$\delta(q_1, b, 1) = \{(q_2, \lambda)\}$
$\delta(q_2, b, 1) = \{(q_2, \lambda)\}$
$\delta(q_2, \lambda, 0) = \{(q_0, \lambda)\}$

خاطر رشته λ ، q_0 حالت نهایی final شده است.

$a^n b^n$ یک زبان مستقل از متن قطعی است زیرا برایش توانستیم DPDA بسازیم.

0 علامت ته stack است.

در خط آخر علامت ته stack را برمی‌دارد تا نتوانند به بالا که بتوانند دوباره a را کدر بردارد.

$\delta(q_2, \lambda, 0) = \{(q_0, \lambda)\}$

و در نتیجه ادامه بدهد همان $a^n b^n$ بماند.

7-4 LL(K) grammars

$S \to ab \mid cd$ → LL(1) (با یک پایانه می‌فهمیم کجا باید برویم و از کدام قانون استفاده کنیم)

$S \to ab \mid ad$ → LL(2) (با دو پایانه می‌فهمیم کجا باید برویم و از کدام قانون استفاده کنیم)

⋮

→ LL(K)

$\begin{cases} S \to A \mid Ab \\ A \to aA \mid \lambda \end{cases}$ → LL(K) نیست

هیچوقت نمی‌توانیم بگوییم که به ازای دیدن چند پایانه بفهمیم که از کدام قانون باید استفاده کنیم.

زبان مستقل از متن قطعی: مستقل از متن که دارای ماشین قطعی داشته باشد. چون می‌شه DPDA برایش نوشت.

غیرقطعی آنی که نشه برایش DPDA نوشت اما شاید برایش nPDA نوشت. مثل زبان $\{ww^R\}$ زبان مستقل از متن غیرقطعی

هر زبان منظم دارای یک گرامر LL(1) است یعنی یک grammar درست کنیم.

زبان مستقل از متن غیرقطعی نمی‌توان گرامرهای LLK نوشت. دلیل نداره که نشه برای همه مستقل از متن های غیرقطعی LLK نوشت.

اگر LLK داره ← حتماً غیرقطعی است.

(7-4 مربوط به درس تصمیم نیست.)

Created with Microsoft Office OneNote 2007
One place for all your notes and information

amitiswebb.com