MODUL 12: BENTUK-BENTUK SEDERHANA DAN BENTUK-BENTUK NORMAL

PENDAHULUAN

Dalam bahasan berikut akan dilakukan cara-cara untuk "memperbaiki" grammar tanpa adanya perubahan penting dari bahasa yang dihasilkannya:

- Dengan menghilangkan **produksi-Λ** dan **produksi unit**
- Dengan pembakuan produksi sehingga mengikuti suatu **Bentuk Normal Chomsky (CNF)**

Produksi-Λ adalah produksi berbentuk $A \to \Lambda$, dan produksi unit adalah produksi berbentuk $A \to B$.

Apabila tidak ada produksi-Λ maka pada setiap kemungkinan penurunan $a \Rightarrow b$ selalu terjadi $|a| \le |b|$. Apabila tidak ada produksi unit maka jika $|a| = |b|$, berarti yang terjadi satu variabel tersubstitusi oleh satu simbol (terminal).

Dalam bentuk grammar demikian maka analisis-analisis teoritis dapat lebih mudah dilakukan. Misalnya, jika l panjang string a dan t jumlah terminal di dalamnya, agar kedua produksi tersebut tidak ada maka $l+t$ harus bertambah pada setiap langkah penurunan. Pada penurunan $S \Rightarrow^* x$, di awal penurunan $l+t = 1$ (untuk S), dan di akhir penurunan $l+t = 2k$ jika $|x| = k$. Jika kedua produksi tersebut tidak ada maka dapat dipastikan paling banyak terjadi $2k$-1 penurunan (Jika lebih banyak dari itu pastilah terjadi salah satu produksi tersebut).

MENGHILANGKAN PRODUKSI-Λ

Ide dasarnya adalah jika dalam grammar terdapat produksi $C \to \Lambda$ maka kemudian kita cari produksi $A \to b$ dimana b berisikan C, lalu, hilangkan $C \to \Lambda$ dan buat produksi baru $A \to b_1$ yaitu b tanpa adanya C. Kalau C muncul lebih dari satu kali di b maka semua produksi berbentuk $A \to b_k$ dibuat dengan b_k merupakan b dengan mensubsitusi secara bergantian setiap C.

Contoh: $S \to CAC$

$A \to Sa$

$C \to \Lambda \mid a$

maka $C \to \Lambda$ dapat dihilangkan disertai dengan menambahkan S → AC (C pertama dihilangkan), $S \to CA$ (C kedua dihilangkan) dan $S \to A$ (kedua C dihilangkan sehingga menjadi

$S \to CAC \mid AC \mid CA \mid A$

$A \to SA$

$C \to a$

Masih terdapat variabel yang potensial menjadi Λ akibat penghilangan variabel tsb misalnya $A \to BC$ sementara $B \to \Lambda$ dan $C \to \Lambda$, maka akan menjadi terbentuk A → Λ. Untuk lebih sistematis pembahasan algoritma penghapusan produksi-Λ ini maka perlu dibuatkan definisi berikut ini.

Variabel *Nullable*

Variabel *nullable* dalam CFG $G = (V, \Sigma, S, P)$ di definisikan sbb.

- variabel A adalah *nullable* jika dalam P terdapat $A \to \Lambda$
- variabel A *nullable* jika dalam P terdapat $A \to B_1B_2\ldots B_n$ dan semua $B_1, B_2, \ldots, B_n$ *nullable*.
- Tidak ada variabel lain yang *nullable*.

Jadi variabel *nullable* adalah variabel yang memiliki, termasuk yang potensial nantinya memiliki, produksi-Λ.

Definisi tersebut dapat diterjemahkan dalam bentuk algoritma pencarian variabel nullable dalam $G = (V, \Sigma, S, P)$ sebagai berikut ini.

Algoritma FindNull

$N_0 = \{A \in V \mid P$ berisikan produksi $A \to \Lambda\}$

do

$i = i+1$

$N_i = N_{i-1} \cup \{A \in V \mid P$ berisikan produksi $A \to a$ untuk $a \in N^*\}$

while $(N_i - N_{i-1}) \neq \varnothing$

Contoh pada grammar

$S \rightarrow ABCE$
$E \rightarrow BDA$
$A \rightarrow CD$
$B \rightarrow Cb$
$C \rightarrow a \mid \Lambda$
$D \rightarrow bD \mid \Lambda$

Mula-mula $N_0 = \{C, D\}$ karena terdapat $C \rightarrow \Lambda$ dan $D \rightarrow \Lambda$, kemudian $N_1 = \{A, C, D\}$ karena terdapat $A \rightarrow CD$, serta $C, D \in N_0$. Karena tidak ada penambahan variabel baru pada N_2 maka himpunan variabel *nullabel* untuk grammar tsb adalah $\{A, C, D\}$.

Algoritma Menghilangkan Produksi-Λ

Dengan algoritma tersebut maka algoritma menghilangkan produksi-Λ dari CFG $G = (V, \Sigma, S, P)$ menjadi CFG $G_1 = (V, \Sigma, S, P_1)$ dapat disusun sebagai berikut.

- Inisialisasi $P_1 = P$
- Mencari variabel-variabel *nullable* dalam V dengan algoritma FindNull
- Untuk setiap produksi $A \rightarrow a$ dalam P tambahkan P_1 setiap bentuk produksi yang dapat dicapai dari produksi ini dengan menghapuskan dari a kemunculan satu atau lebih variabel nullable.
- Hapus semua produksi-Λ dari P_1
- Jika ada hapus produksi berbentuk $A \rightarrow A$
- Jika ada hapus duplikasi (beberapa produksi yang sama ambil satu dan hapus yang lain).

Dari contoh di sebelumnya setelah mendapatkan variabel *nullable* $\{A, C, D\}$ maka sejumlah aturan produksi baru ditambahkan sbb.
Pada $S \rightarrow ABCE$ terdapat A dan C, sehingga $S \rightarrow ABCE \mid BCE \mid ABE \mid BE$
Pada $E \rightarrow BDA$ terdapat D dan A, sehingga $E \rightarrow BDA \mid DA \mid BD \mid D$
Pada $A \rightarrow CD$ terdapat C dan D, sehingga $A \rightarrow CD \mid C \mid D \mid \Lambda$
Dan seterusnya. Setelah menghilangkan semua produksi-Λ maka diperoleh

$S \rightarrow ABCE \mid BCE \mid ABE \mid BE$
$E \rightarrow BDA \mid DA \mid BD \mid D$
$A \rightarrow CD \mid C \mid D$
$B \rightarrow Cb \mid b$
$C \rightarrow a$
$D \rightarrow bD$

Jika S adalah *nullable* maka grammar G_1 yang dihasilkan algoritma di atas tidak persis sama dengan grammar G semula karena $L(G_1) = L(G) - \{\Lambda\}$. Untuk menjaga bahwa grammar tetap mendefinisikan grammar yang sama, jika S adalah nullable maka $S \rightarrow \Lambda$ dalam P_1 (jadi terdapat pengecualian adanya produksi-Λ untuk S).

Tanpa itu, maka secara umum dapat dikatakan bahwa $L(G_1) = L(G) - \{\Lambda\}$.

Untuk contoh di atas karena S bukan nullable maka grammar yang dihasilkan mendefinisikan bahasa yang tepat sama dengan yang didefinisikan grammar semula.

MENGHILANGKAN PRODUKSI UNIT

Ide dasar: untuk menghilangkan produksi $A \rightarrow B$ sementara terdapat $B \rightarrow a$ maka dibuat $A \rightarrow a$ sebagai gantinya. Namun, karena bisa terdapat rantai produksi unit $A \rightarrow B$, $B \rightarrow C$, … maka perlu dibuatkan terlebih dahulu definisi berikut ini.

Derrivabilitas

Pada suatu CFG $G = \{V, \Sigma, S, P\}$, dan setiap $A, B, C \in V$

- Jika $A \rightarrow B$ suatu produksi dalam P maka B disebut *A-derrivable*.
- Jika C adalah *A-derrivable*, dan jika $C \rightarrow B$ suatu produksi dalam P maka B juga *A-derrivable*.
- Tidak ada variabel lain dalam V yang *A-derrivable*.

Contoh pada grammar

$S \rightarrow S+T \mid T$
$T \rightarrow T*F \mid F$
$F \rightarrow (S) \mid a$

Maka T dan F adalah *S-derivable*.

Dengan definisi tersebut maka algoritma penghilangan produksi unit dapat disusun sebagai berikut.

Algoritma Menghilangkan Produksi Unit

Dari grammar $G = (V, \Sigma, S, P)$ yang tidak memiliki produksi-Λ, grammar $G_1 = (V, \Sigma, S, P_1)$ yang tidak memiliki produksi unit dapat dibuat sebagai berikut.

1. Inisialisasi P_1 dengan P.
2. Untuk setiap $A \in V$ temukan *A-derrivable*.
3. Untuk setiap pasangan (A, B) dimana B adalah *A-derrivable* dan setiap produksi $B \rightarrow a$ yang bukan produksi unit, tambahkan $A \rightarrow a$ dalam P_1.
4. Hapus semua produksi unit dari P_1.

Perhatikan bahwa algoritma ini mensyaratkan grammar G sudah tidak memiliki produksi-Λ. Untuk grammar sembarang maka sebelum dihilangkan produksi unitnya, perlu di hilangkan dulu produksi-Λ-nya.

Mengingat bahwa setelah produksi-Λ dihilangkan berlaku $L(G_1) = L(G) - \{\Lambda\}$, maka setelah penghilangan produksi unit sifat itu juga tetap berlaku.

Untuk contoh sebelumnya himpunan *S-derrivable* = {*F*, *T*}, himpunan *T-derivable* = {*T*}, himpunan *F-derrivable* = {}. Dari langkah ke-3 diperoleh

$S \rightarrow S+T \mid T \mid T*F \mid F \mid (S) \mid a$
$T \rightarrow T*F \mid F \mid (S) \mid a$
$F \rightarrow (S) \mid a$

Dari langkah-4, maka tersisa aturan-aturan produksi:

$S \rightarrow S+T \mid T*F \mid (S) \mid a$
$T \rightarrow T*F \mid (S) \mid a$
$F \rightarrow (S) \mid a$

BENTUK NORMAL CHOMSKY (CNF)

Seperti pada pembahasan di awal bahwa penghilangan kedua macam produksi yang telah dibahas adalah untuk mendapatkan grammar yang lebih mudah dianalisis secara teoritis. Hal ini ditunjukkan dengan jumlah langkah pada penurunan $S \Rightarrow^* x$ yang tidak akan lebih banyak dari $2|x| - 1$. Selain batas atas jumlah langkah penurunan, juga terkadang batas bawah jmlah penurunan diperlukan, bahkan kalau bisa jumlah langkah adalah fungsi dari $|x|$. Bentuk normal Chomsky adalah bentuk grammar yang memastikan bahwa jumlah langkah penurunan pada $S \Rightarrow^* x$ tepat $2|x| - 1$.

Definisi CNF

Suatu CNF G berada dalam bentuk normal Chomsky (CNF/*Chomsky Normal Form*) apabila setiap produksi mengambil salah satu dari dua pola:

$A \rightarrow BC$
$A \rightarrow a$

Dimana A, B, C adalah variabel dan a adalah simbol.

Algoritma Transformasi ke CNF

Sebelumnya perlu dilakukan penghapusan produksi-Λ dan produksi unit terlebih dahulu pada grammar G yang akan ditransformasikan. Selanjutnya setelah diperoleh grammar G_1 tanpa kedua produksi tersebut, $G_2 = (V, \Sigma, S, P_2)$ diperoleh dalam dua tahap.

1. Setiap produksi dibuat untuk berada dalam format $A \rightarrow B_1B_2 \ldots B_k$, dengan $k \geq 2$ dan $B_i \in V_2$, atau $A \rightarrow a$, untuk $a \in \Sigma$.
   Untuk itu, jika terdapat $A \rightarrow a$ dan $|a| \geq 2$ di dalam a terdapat simbol a maka dibuat suatu variabel baru misalnya X_a dan $X_a \rightarrow a$ ke dalam P_2 serta mengganti setiap a dalam a dengan X_a. Kecuali, kalau sebelumnya sudah ada produksi $A \rightarrow a$, maka X_a dan $X_a \rightarrow a$ tidak perlu dibuat, tapi setiap a dalam a diganti A.

2. Setiap produksi $A \rightarrow B_1B_2B_3 \ldots B_{k-1}B_k$ dengan $k > 2$ diganti dengan sejumlah produksi yang ekivalen yang masing-masing berbentuk $X \rightarrow YZ$. Penggantian yang bisa dilakukan adalah dengan mengganti $B_2B_3 \ldots B_k$ dengan variabel baru X_1 dan menambahkan $X_1 \rightarrow B_2X_2$, $X_2 \rightarrow X_3B_3$, $\ldots$ $X_{k-2} \rightarrow B_{k-1}B_k$.

Untuk contoh sebelumnya setelah menghilangkan produksi unit, grammar:

$S \rightarrow S+T \mid T*F \mid (S) \mid a$

$T \rightarrow T*F \mid (S) \mid a$
$F \rightarrow (S) \mid a$

dapat dikonversi ke dalam CNF sbb. Step 1 menghasilkan

$S \rightarrow SX_+T \mid TX_*F \mid X_(SX_) \mid a$
$T \rightarrow TX_*F \mid X_(SX_) \mid a$
$F \rightarrow X_(SX_) \mid a$
$X_+ \rightarrow +$
$X_* \rightarrow *$
$X_(\rightarrow ($
$X_) \rightarrow)$

Step 2 menghasilkan

$S \rightarrow SZ_1 \mid TZ_2 \mid X_3Z_3 \mid a$
$T \rightarrow TZ_2 \mid X_3Z_3 \mid a$
$F \rightarrow X_3Z_3 \mid a$
$Z_1 \rightarrow X_1T$
$Z_2 \rightarrow X_2F$
$Z_3 \rightarrow SX_4$
$X_1 \rightarrow +$
$X_2 \rightarrow *$
$X_3 \rightarrow ($
$X_4 \rightarrow)$

Yang sudah dalam CNF.

Mengingat bahwa setelah produksi-Λ dihilangkan dari G menjadi G_1, berlaku $L(G_1) = L(G) - \{\Lambda\}$, maka setelah dalam CNF menjadi G_2 sifat itu juga tetap berlaku. Yang artinya, jika $\Lambda \in L(G)$ dan G grammar semula dikonversi ke dalam CNF menjadi G_2 maka $\Lambda \notin L(G_2)$.

Contoh: diberikan CFG G dengan produksi-produksi

$S \rightarrow AACD$
$A \rightarrow aAb \mid \Lambda$
$C \rightarrow aC \mid a$
$D \rightarrow aDa \mid bDb \mid \Lambda$

Variabel-variabel nullable adalah A, D. Proses mengilangkan produksi-Λ pada G menghasilkan G_1 dengan produksi-produksi:

$S \rightarrow AACD \mid ACD \mid AAC \mid CD \mid AC \mid C$
$A \rightarrow aAb \mid ab$
$C \rightarrow aC \mid a$
$D \rightarrow aDa \mid bDb \mid aa \mid bb$

Penghilangan produksi unit pada G_1 menghasilkan G_2 dengan produksi-produksi:

$S \rightarrow AACD \mid ACD \mid AAC \mid CD \mid AC \mid aC \mid a$
$A \rightarrow aAb \mid ab$
$C \rightarrow aC \mid a$
$D \rightarrow aDa \mid bDb \mid aa \mid bb$

Tahap pertama konversi ke CNF menghasilkan G_3 dengan produksi-produksi:

$S \rightarrow AACD \mid \mathrm{ACD} \mid AAC \mid CD \mid AC \mid X_aC \mid a$
$A \rightarrow X_aAX_b \mid X_aX_b$
$C \rightarrow X_aC \mid a$
$D \rightarrow X_aDX_a \mid X_bDX_b \mid X_aX_a \mid X_bX_b$
$X_a \rightarrow a$
$X_b \rightarrow b$

Tahap kedua konversi ke CNF menghasilkan G_4 dengan prduksi-produksi:

$S \rightarrow AT_1 \mid AU_1 \mid AV_1 \mid CD \mid AC \mid X_aC \mid a$
$A \rightarrow X_aW_1 \mid X_aX_b$
$C \rightarrow X_aC \mid a$
$D \rightarrow X_aY_1 \mid X_bZ_1 \mid X_aX_a \mid X_bX_b$
$X_a \rightarrow a$
$X_b \rightarrow b$
$T_1 \rightarrow AT_2$
$T_2 \rightarrow CD$

$U_1 \rightarrow CD$
$V_1 \rightarrow AC$
$W_1 \rightarrow AX_b$
$Y_1 \rightarrow DX_a$
$Z_2 \rightarrow DX_b$