Simplifications of Context-Free Grammars

class 9

1

A Substitution Rule

$S \to aB$
$A \to aaA$
$A \to abBc$
$B \to aA$
$B \to b$

Substitute $B \to b$

Equivalent grammar

$S \to aB \mid ab$
$A \to aaA$
$A \to abBc \mid abbc$
$B \to aA$

2

A Substitution Rule

$S \to aB \mid ab$
$A \to aaA$
$A \to abBc \mid abbc$
$B \to aA$

Substitute $B \to aA$

$S \to \cancel{aB} \mid ab \mid aaA$
$A \to aaA$
$A \to \cancel{abBc} \mid abbc \mid abaAc$

Equivalent grammar

3

In general:

$A \to xBz$

$B \to y_1$

Substitute $B \to y_1$

$A \to xBz \mid xy_1z$

equivalent grammar

4

Nullable Variables

λ – production: $A \to \lambda$

Nullable Variable: $A \Rightarrow \ldots \Rightarrow \lambda$

5

Removing Nullable Variables

Example Grammar:

$S \to aMb$
$M \to aMb$
$M \to \lambda$

Nullable variable

6

Final Grammar

$$S \to aMb$$
$$M \to aMb$$
$$\cancel{M \to \lambda}$$

Substitute $M \to \lambda$

$$S \to aMb$$
$$S \to ab$$
$$M \to aMb$$
$$M \to ab$$

7

Unit-Productions

Unit Production: $A \to B$

(a single variable in both sides)

8

Removing Unit Productions

Observation:

$$A \to A$$

Is removed immediately

9

Example Grammar:

$$S \to aA$$
$$A \to a$$
$$A \to B$$
$$B \to A$$
$$B \to bb$$

10

$$S \to aA$$
$$A \to a$$
$$\cancel{A \to B}$$
$$B \to A$$
$$B \to bb$$

Substitute $A \to B$

$$S \to aA \mid aB$$
$$A \to a$$
$$B \to A \mid B$$
$$B \to bb$$

11

$$S \to aA \mid aB$$
$$A \to a$$
$$B \to A \mid \cancel{B}$$
$$B \to bb$$

Remove $B \to B$

$$S \to aA \mid aB$$
$$A \to a$$
$$B \to A$$
$$B \to bb$$

12

$S \to aA \mid aB$

$A \to a$

$\cancel{B \to A}$

$B \to bb$

Substitute
$B \to A$

$S \to aA \mid aB \mid aA$

$A \to a$

$B \to bb$

13

Remove repeated productions

$S \to aA \mid aB \mid \cancel{aA}$

$A \to a$

$B \to bb$

Final grammar

$S \to aA \mid aB$

$A \to a$

$B \to bb$

14

Useless Productions

$S \to aSb$

$S \to \lambda$

$S \to A$

$A \to aA$ Useless Production

Some derivations never terminate...

$S \Rightarrow A \Rightarrow aA \Rightarrow aaA \Rightarrow \ldots \Rightarrow aa\ldots aA \Rightarrow \ldots$

15

Another grammar:

$S \to A$

$A \to aA$

$A \to \lambda$

$B \to bA$ Useless Production

Not reachable from S

16

In general:

contains only terminals

if $S \Rightarrow \ldots \Rightarrow xAy \Rightarrow \ldots \Rightarrow w$

$w \in L(G)$

then variable A is useful

otherwise, variable A is useless

17

A production $A \to x$ is useless if any of its variables is useless

$S \to aSb$

$S \to \lambda$

Productions

Variables

$S \to A$ useless

useless $A \to aA$ useless

useless $B \to C$ useless

useless $C \to D$ useless

18

Removing Useless Productions

Example Grammar:

$$S \to aS \mid A \mid C$$
$$A \to a$$
$$B \to aa$$
$$C \to aCb$$

First: find all variables that can produce strings with only terminals

$$S \to aS \mid A \mid C$$
$$A \to a$$
$$B \to aa$$
$$C \to aCb$$

Round 1: $\{A, B\}$

$S \to A$

Round 2: $\{A, B, S\}$

Keep only the variables that produce terminal symbols: $\{A, B, S\}$

(the rest variables are useless)

$$S \to aS \mid A \mid \cancel{C}$$
$$A \to a$$
$$B \to aa$$
$$\cancel{C \to aCb}$$

$\Rightarrow$

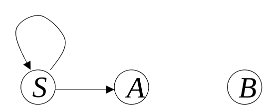
$$S \to aS \mid A$$
$$A \to a$$
$$B \to aa$$

Remove useless productions

Second: Find all variables reachable from S

Use a Dependency Graph

$$S \to aS \mid A$$
$$A \to a$$
$$B \to aa$$



not reachable

Keep only the variables reachable from S

(the rest variables are useless)

$$S \to aS \mid A$$
$$A \to a$$
$$\cancel{B \to aa}$$

$\Rightarrow$

Final Grammar

$$S \to aS \mid A$$
$$A \to a$$

Remove useless productions

Removing All

Step 1: Remove Nullable Variables

Step 2: Remove Unit-Productions

Step 3: Remove Useless Variables

Normal Forms
for
Context-free Grammars

Chomsky Normal Form

Each productions has form:

$$A \to BC \qquad \text{or} \qquad A \to a$$

variable variable terminal

Examples:

$$S \to AS$$
$$S \to a$$
$$A \to SA$$
$$A \to b$$

Chomsky
Normal Form

$$S \to AS$$
$$S \to AAS$$
$$A \to SA$$
$$A \to aa$$

Not Chomsky
Normal Form

Convertion to Chomsky Normal Form

Example:

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

Not Chomsky
Normal Form

Introduce variables for terminals: T_a, T_b, T_c

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

$$S \to ABT_a$$
$$A \to T_aT_aT_b$$
$$B \to AT_c$$
$$T_a \to a$$
$$T_b \to b$$
$$T_c \to c$$

Introduce intermediate variable: V_1

$$S \to ABT_a$$
$$A \to T_aT_aT_b$$
$$B \to AT_c$$
$$T_a \to a$$
$$T_b \to b$$
$$T_c \to c$$

$$S \to AV_1$$
$$V_1 \to BT_a$$
$$A \to T_aT_aT_b$$
$$B \to AT_c$$
$$T_a \to a$$
$$T_b \to b$$
$$T_c \to c$$

Introduce intermediate variable: V_2

$$S \to AV_1 \quad V_1 \to BT_a \quad A \to T_aT_aT_b \quad B \to AT_c \quad T_a \to a \quad T_b \to b \quad T_c \to c$$

$$S \to AV_1 \quad V_1 \to BT_a \quad A \to T_aV_2 \quad V_2 \to T_aT_b \quad B \to AT_c \quad T_a \to a \quad T_b \to b \quad T_c \to c$$

31

Final grammar in Chomsky Normal Form:

Initial grammar

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

$$S \to AV_1 \quad V_1 \to BT_a \quad A \to T_aV_2 \quad V_2 \to T_aT_b \quad B \to AT_c \quad T_a \to a \quad T_b \to b \quad T_c \to c$$

32

In general:

From any context-free grammar
(which doesn't produce λ)
not in Chomsky Normal Form

we can obtain:
An equivalent grammar
in Chomsky Normal Form

33

The Procedure

First remove:

Nullable variables

Unit productions

34

Then, for every symbol a:

Add production $T_a \to a$

In productions: replace a with T_a

New variable: T_a

35

Replace any production $A \to C_1C_2 \cdots C_n$

with
$$A \to C_1V_1$$
$$V_1 \to C_2V_2$$
$$\cdots$$
$$V_{n-2} \to C_{n-1}C_n$$

New intermediate variables: $V_1, V_2, \ldots, V_{n-2}$

36

Theorem: For any context-free grammar (which doesn't produce λ) there is an equivalent grammar in Chomsky Normal Form

37

Observations

- Chomsky normal forms are good for parsing and proving theorems

- It is very easy to find the Chomsky normal form for any context-free grammar

38

Greinbach Normal Form

All productions have form:

$$A \to a\, V_1 V_2 \cdots V_k \qquad k \geq 0$$

symbol variables

39

Examples:

$$S \to cAB$$
$$A \to aA \mid bB \mid b$$
$$B \to b$$

Greinbach Normal Form

$$S \to abSb$$
$$S \to aa$$

Not Greinbach Normal Form

40

Conversion to Greinbach Normal Form:

$$S \to abSb$$
$$S \to aa$$

$$S \to aT_bST_b$$
$$S \to aT_a$$
$$T_a \to a$$
$$T_b \to b$$

Greinbach Normal Form

41

Theorem: For any context-free grammar (which doesn't produce λ) there is an equivalent grammar in Greinbach Normal Form

42

Observations

- Greinbach normal forms are very good for parsing

- It is hard to find the Greinbach normal form of any context-free grammar

43

The CYK Parser

44

The CYK Membership Algorithm

Input:

- Grammar G in Chomsky Normal Form
- String w

Output:

find if $w \in L(G)$

45

The Algorithm

Input example:

- Grammar G: $S \to AB$
  $A \to BB$
  $A \to a$
  $B \to AB$
  $B \to b$

- String w: $aabbb$

46

$aabbb$

a	a	b	b	b
aa	ab	bb	bb	
aab	abb	bbb		
aabb	abbb			
aabbb				

47

$S \to AB$
$A \to BB$
$A \to a$
$B \to AB$
$B \to b$

a A	a A	b B	b B	b B
aa	ab	bb	bb	
aab	abb	bbb		
aabb	abbb			
aabbb				

48

$S \to AB$
$A \to BB$
$A \to a$
$B \to AB$
$B \to b$

a A	a A	b B	b B	b B
aa	ab S,B	bb A	bb A	
aab	abb	bbb		
aabb	abbb			
aabbb				

49

$S \to AB$
$A \to BB$
$A \to a$
$B \to AB$
$B \to b$

a A	a A	b B	b B	b B
aa	ab S,B	bb A	bb A	
aab S,B	abb A	bbb S,B		
aabb A	abbb S,B			
aabbb S,B				

50

Therefore: $aabbb \in L(G)$

Time Complexity: $|w|^3$

Observation: The CYK algorithm can be easily converted to a parser (bottom up parser)

51