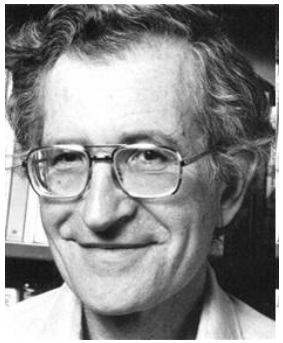
Context Free Grammar (CFG)

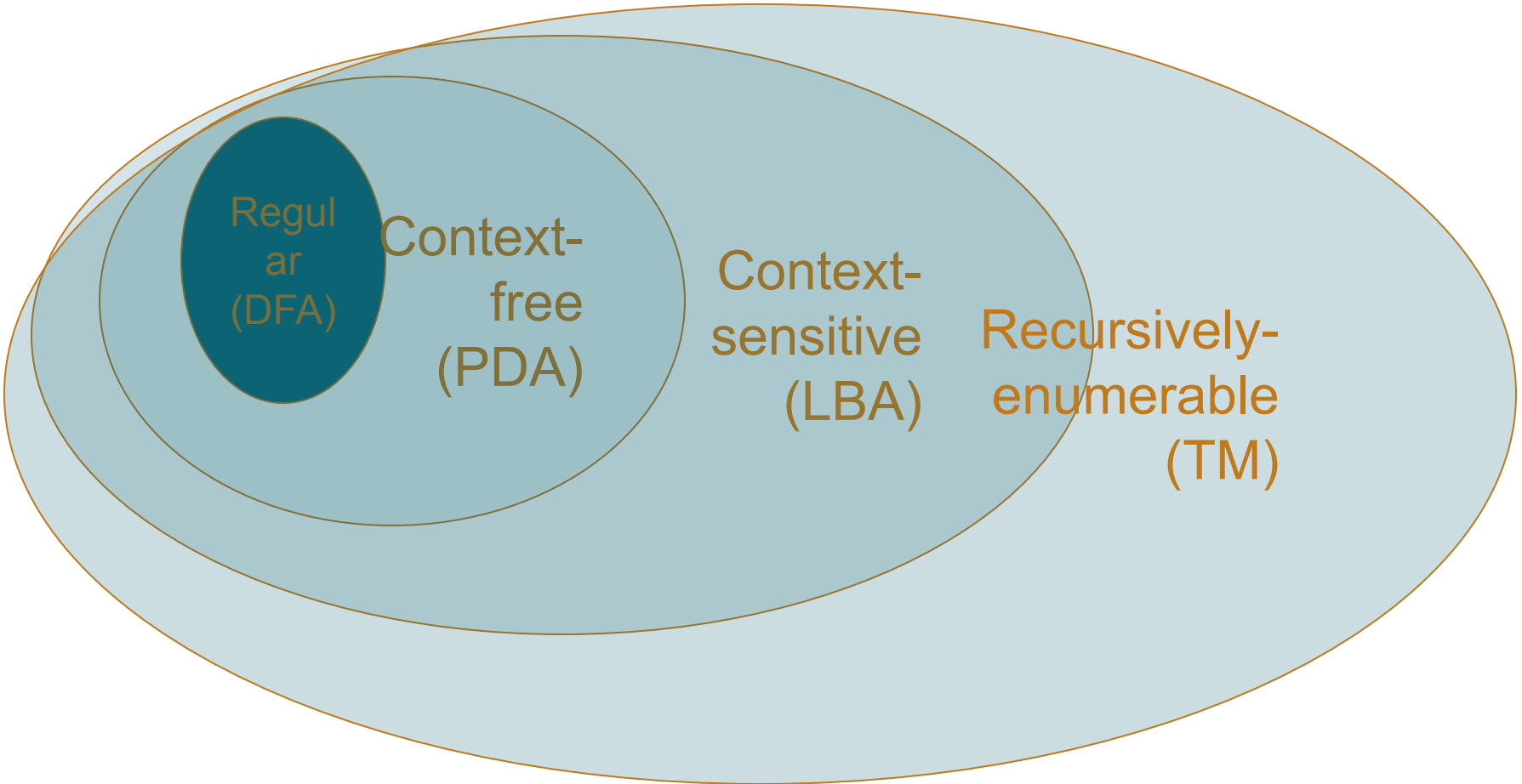
By

NAYAN KUMAR

The Chomsky Hierachy



- A containment hierarchy of classes of formal languages



Formal Definition

One can provide a formal definition of a context free grammar. It is a 4-tuple (V, T, S, P) where:

- V is a finite set of variables;
- T is a finite alphabet of terminals;
- S is the start variable; and
- P is the finite set of productions. Each production has the form $V \rightarrow (V \cup \Sigma)^*$.

Example-1

Find CFG for L1 = { a^nb^n | n is a positive integer }

Let the CFG is G = < V ,T , P , S>
V = { X } ,
T = { a , b } and
P = { X -> aXb , X -> ab }
S = X.

Example-2

Find CFG for L2 = { ww^r| w ∈ {a, b}+ }

Let the CFG is G = < V ,T , P , S>
V = { S } ,
T = { a , b } and
P = { S -> aSa , S -> bSb , S -> aa , S -> bb }.
S = S

Example-3

Write a CFG, which generates palindrome for binary numbers.

The grammar will generate palindrome for binary numbers, that is 00, 010, 001100, …

Let G be the CFG G = <V, T, P, S>
V = {S}
T = {0, 1}
And production rule P is defined as
S -> 0S0/ 1S1
S -> 0/1/ϵ

Example-4

Write a CFG for language L = {0^i1^j2^k | k <= i or k <= j}

We can think of L as union of two language,
L = L1 U L2
Where L1 = {0i1j2k | k <= i} And L1 = {0^i1^j2^k | k <= j}

Let us assume that CFG for L1 is G1
G1 = (V1, T1, P1, S1)
V1 = {X, Y, Z}
T1 = {0, 1}
And P1 is defined as
X -> X2/0Y2
Y -> Z/0Y/ϵ
Z -> 1Z/ϵ

S1=X

Example-4 [Cont...]

Let us assume that CFG for L2 is G2
G2 = (V2, V2, P2, S2)
V2 = {C, A, B}
T2 = {0, 1}
S2 = C
And P2 is defined as
C -> AB
A -> 0A/ϵ
B -> 1B2/1B/ϵ

Example-4 [Cont...]

We can define CFG for L(since L = L1 U L2)
Let us assume that CFG for L is G
G = (V, T, P, S)
V = { X, Y, Z, A, B, C}
T = {0, 1}
S = S

P is defined as
S -> X/C
X -> X2/0Y2
Y -> Z/0Y/ϵ
Z -> 1Z/ϵ
C -> AB
A -> 0A/ϵ
B -> 1B2/1B/ϵ

Example-5

Write a CFG for the language L = {wcwr | w ϵ (a, b)*}

Let us check that "**abbcbba**" can be derived from the given CFG.

S =>aSa [use the production S -> aSa]
=> abSba [use S -> bSb]
=> abbSbba [use S ->bSb]
=> abbcbba [use S->c]
So string "**abbcbba**" can be derived from given CFG.

Let G be a CFG for the language L. G = (V, T, P,S) Here,
V = {S}
T = {a, b, c}
And P is given by
S -> aSa
S -> bSb
S -> c

Derivation

Derivation is a sequence of production rules.It is used to get the input string through these production rules.

During parsing, we have to take two decisions. These are as follows:

1. We have to decide the non-terminal which is to be replaced.
2. We have to decide the production rule by which the non-terminal will be replaced. We have two options to decide which non-terminal to be placed with production rule.

Leftmost Derivation:

In the leftmost derivation, the input is scanned and replaced with the production rule from left to right. So in leftmost derivation, we read the input string from left to right.

Example: Production rules: E -> E + E

E -> E - E

E -> a | b

Input 1. a - b + a

The leftmost derivation is: 1. E = E + E

2. E = E - E + E

3. E = a - E + E

4. E = a - b + E

5. E = a - b + a

Rightmost Derivation:

In rightmost derivation, the input is scanned and replaced with the production rule from right to left. So in rightmost derivation, we read the input string from right to left

Example: Production rules: E -> E + E

E -> E - E

E -> a | b

Input 1. a - b + a

The rightmost derivation is: 1. E = E - E

2. E = E - E + E

3. E = E - E + a

4. E = E - b + a

5. E = a - b + a

Examples of Derivation:

Example-1:

Derive the string **"abb"** for leftmost derivation and rightmost derivation using a CFG given by,

1. $S \rightarrow AB \mid \varepsilon$
2. $A \rightarrow aB$
3. $B \rightarrow Sb$

Example-1:

Leftmost derivation:

S

AB

aB B

a Sb B

A ε bB

ab Sb

ab ε b

abb

Rightmost derivation:

S

AB

A Sb

A ε b

aB b

a Sb b

a ε bb

abb

Examples of Derivation:

Example-2:

Derive the string **"aabbabba"** for leftmost derivation and rightmost derivation using a CFG given by,

1. S → aB | bA
2. S → a | aS | bAA
3. S → b | aS | aBB

Example-2:

Leftmost derivation:

1. S	
2. aB	S → aB
3. aaBB	B → aBB
4. aabB	B → b
5. aabbS	B → bS
6. aabbaB	S → aB
7. aabbabS	B → bS
8. aabbabbA	S → bA
9. aabbabba	A → a

Rightmost derivation:

1. S	
2. aB	S → aB
3. aaBB	B → aBB
4. aaBbS	B → bS
5. aaBbbA	S → bA
6. aaBbba	A → a
7. aabSbba	B → bS
8. aabbAbba	S → bA
9. aabbabba	A → a

Examples of Derivation: :

Example-3:

Derive the string **"00101"** for leftmost derivation and rightmost derivation using a CFG given by,

1. $S \rightarrow A1B$
2. $A \rightarrow 0A \mid \varepsilon$
3. $B \rightarrow 0B \mid 1B \mid \varepsilon$

Example-3:

Leftmost derivation:

1. S
2. A1B
3. 0A1B
4. 00A1B
5. 001B
6. 0010B
7. 00101B
8. 00101

Rightmost derivation:

1. S
2. A1B
3. A10B
4. A101B
5. A101
6. 0A101
7. 00A101
8. 00101