Formal Grammar

Amba Kulkarni

Fellow IIAS, Shimla

8th April 2017

- Language: A collection of sentences of finite length constructed from a finite alphabet of symbols
- Grammar: A device that enumerates the sentences of a language
  Grammar of a language L is a function whose range is L.

Types of Grammars

- Prescriptive: Prescribes authoritative norms for a language
- Descriptive: Describes the actual usages
- Formal: A grammar that can be executed by formal devices such as Automaton

Types of Grammars

Formal grammar can be

- Generative: A formal grammar that can 'generate' natural language expressions
- Analytic: Given a linguistic expression, this grammar checks the validity of the input and analyses it into meaningful ints showing the inter-relations.

Generative grammar: Gives a procedure for 'writing' correct sentences in a language L

Analytic Grammar: Gives a procedure for 'analysing' the sentences in a langauge L.

Formal Grammar

$G = (N, T, P, S)$

- N : A finite set of Non-terminals
- S: The Start Symbol $\in N$
- T: A finite set of Terminals
- P: Production rules, of the type
  $(T \cup N) * N(T \cup N)* \rightarrow (T \cup N)*$

Chomsky's Hierarchy

- Regular languages
- Context Free languages
- Context Sensitive languages
- Recursively enumerable languages

Chomsky's Hierarchy

Class	Grammars	Languages	Automaton
Type-0	Unrestricted	Recursively Enumerable	TM
Type-1	Context Sensitive	Context Sensitive	LBA
Type-2	Context Free	Context Free	PA
Type-3	Regular	Regular	FSA

FA: Finite State Automaton
PA: Pushdown Automaton
LBA: Linear Bound Automaton
TM: Turing Machine

Chomsky's Hierarchy

Type 3 : Regular

Languages defined by Type-3 grammars are accepted by Finite Automata

Rules are of the form:
$A \rightarrow \epsilon$,
$A \rightarrow \alpha$,
$A \rightarrow \alpha B$
OR
$A \rightarrow \epsilon$,
$A \rightarrow \alpha$,
$A \rightarrow B\alpha$
where $A, B \in N, and \alpha \in T$
The first one yields a right branching parse tree while the second one a left branching parse tree.

Example

A language $\{a^n \ / \ n \geq 0\}$ is an example of Type-3 / regular language.

$G = (N, T, P, S)$

- N : A finite set of Non-terminals
- S: The Start Symbol $\in N$
- T: A finite set of Terminals
- P: Production rules, of the type

Let N = {S}
Let T = {a}
Let P:
S := a S
| ϵ

This grammar generates a language:
ϵ, a, aa, aaa, aaaa, ...

Chomsky's Hierarchy

Type 2 : Context Free

Languages defined by Type-2 grammars are accepted by Pushdown Automata

Rules are of the form:
$A \rightarrow \alpha$
where $A \in N$
$\alpha \in (N \cup T)*$

A FSA can not recognise strings from this type of language since we must be able to "remember" the information. PDA which uses stack can recognise such strings.

Almost all natural languages fall under this category.
Programming languages fall under this category (most part of it).

Example

A language $\{0^n1^n \ / \ n \geq 0\}$ is an example of Type-2 / regular language.

$G = (N, T, P, S)$

- N : A finite set of Non-terminals
- S: The Start Symbol $\in N$
- T: A finite set of Terminals
- P: Production rules, of the type

Let N = {S}
Let T = {0,1}
Let P:
S := 0S1
| ϵ

This grammar generates a language:
ϵ, 01, 0011, 000111, 00001111, ...

Chomsky's Hierarchy

Type 1 : Context Sensitive

Languages defined by Type-1 grammars are accepted by Linear Bound Automata

Rules are of the form:
$\alpha A \beta \rightarrow \alpha B \beta$ and
$S \rightarrow \epsilon$ where $A, S \in N$
α, β, B $\in (N \cup T)*$

The constructions with 'respectively' fall under this category.
Germanic languages are also context sensitive.

Example

A language $\{a^n b^n c^n \ / \ n \geq 0\}$ is an example of Type-2 / regular language.

Let N = {S}
Let T = {a,b,c}
Let P:
S := aSBC
| aBC
| ϵ
CB := BC
aB := ab
bB := bb
bC := bc
cC := cc

This grammar generates a language:
ϵ, abc, aabbcc, aaabbbccc, ...

Chomsky's Hierarchy

Type 0 (Unrestricted)

Languages defined by Type-0 grammars are accepted by Turing Machines.
Rules are of the form:
$\alpha \rightarrow \beta$,
where α and β are arbitraty strings over a vocabulary V and $\alpha \neq \epsilon$