Natural Language Processing
- Formal Language -

(formal) Language

(formal) Grammar

Formal Language

A formal language L is a set of finite-length words (or "strings") over some finite alphabet A. ε is the empty word.

Example:

A = {a, b, c}

L_1 = {ab, c}

Formal Languages - Examples

Some examples of formal languages:

- the set of all words over {a, b},
- the set { a^n | n is a prime number },
- the set of syntactically correct programs in some programming language, or
- the set of inputs upon which a certain Turing machine halts.

Several operations can be used to produce new languages from given ones. Suppose $L1$ and $L2$ are languages over some common alphabet.

- The *concatenation* $L1L2$ consists of all strings of the form vw where v is a string from $L1$ and w is a string from $L2$.
- The *intersection* of $L1$ and $L2$ consists of all strings which are contained in $L1$ and also in $L2$.
- The *union* of $L1$ and $L2$ consists of all strings which are contained in $L1$ or in $L2$.
- The *complement* of the language $L1$ consists of all strings over the alphabet which are not contained in $L1$.
- The *Kleene star* $L1^*$ consists of all strings which can be written in the form $w1w2...wn$ with strings wi in $L1$ and $n \geq 0$. Note that this includes the empty string ε because $n = 0$ is allowed.

A formal language can be specified in a great variety of ways, such as:

- Strings produced by some formal grammar (see Chomsky hierarchy)
- Strings produced by a regular expression
- Strings accepted by some automaton, such as a Turing machine or finite state automaton
- From a set of related YES/NO questions those ones for which the answer is YES, see decision problem

Formal Grammar - Definition

A formal grammar G = (N, Σ, P, S) consists of:

- A finite set *N* of nonterminal symbols.
- A finite set Σ of terminal symbols that is disjoint from *N*.
- A finite set *P* of production rules where a rule is of the form
    - string in (Σ U *N*)* -> string in (Σ U *N*)*
  - (where * is the Kleene star and U is set union)
  - the left-hand side of a rule must contain at least one nonterminal symbol.
- A symbol *S* in *N* that is indicated as the start symbol.

Language of a Formal Grammar

The language of a formal grammar $G = (N, \Sigma, P, S)$, denoted as $\mathbf{L}(G)$, is defined as all those strings over Σ that can be generated by starting with the start symbol *S* and then applying the production rules in *P* until no more nonterminal symbols are present.

Language of a Formal Grammar

Example

Consider, for example, the grammar G with $N = \{S, B\}$, $\Sigma = \{a, b, c\}$, P consisting of the following production rules

1. S -> aBSc
2. S -> abc
3. Ba -> aB
4. Bb -> bb

This grammar defines the language $\{a^n b^n c^n \mid n>0\}$

Chomsky's four types of grammars

- Type-0 grammars (unrestricted grammars)

  languages recognized by a Turing machine
- Type-1 grammars (context-sensitive grammars)

  Turing machine with bounded tape
- Type-2 grammars (context-free grammars)

  non-deterministic pushdown automaton
- Type-3 grammars (regular grammars)

  regular expressions, finite state automaton

Grammars, Languages, Machines

Type-0		
Recursively enumerable	Turing machine	No restrictions
Type-1		
Context-sensitive	Linear-bounded non-deterministic Turing machine	$\alpha A \beta \rightarrow \alpha \gamma \beta$
Type-2		
Context-free	Non-deterministic pushdown automaton	$A \rightarrow \gamma$
Type-3		
Regular	Finite state automaton	$A \rightarrow aB$ $A \rightarrow a$

Example

EXAMPLE 4.1

$G = (V_N, \Sigma, P, S)$ is a grammar

where

$V_N = \{\langle \text{sentence} \rangle, \langle \text{noun} \rangle, \langle \text{verb} \rangle, \langle \text{adverb} \rangle\}$

$\Sigma = \{\text{Ram, Sam, ate, sang, well}\}$

$S = \langle \text{sentence} \rangle$

P consists of the following productions:

$\langle \text{sentence} \rangle \rightarrow \langle \text{noun} \rangle \langle \text{verb} \rangle$

$\langle \text{sentence} \rangle \rightarrow \langle \text{noun} \rangle \langle \text{verb} \rangle \langle \text{adverb} \rangle$

$\langle \text{noun} \rangle \rightarrow \text{Ram}$

$\langle \text{noun} \rangle \rightarrow \text{Sam}$

$\langle \text{verb} \rangle \rightarrow \text{ate}$

$\langle \text{verb} \rangle \rightarrow \text{sang}$

$\langle \text{adverb} \rangle \rightarrow \text{well}$