Alphabets, words, and languages

Chapitre 01

Alphabet

- A non-empty finite set of symbols (letters)
- An alphabet will, for example, be denoted X or Σ.

- For example

- $\Sigma = \{0,1\}$: alphabet of binary numbers.
- $\Sigma = \{a,b,\ldots,z\}$: the set of lowercase letters.

Word

- A word (finite) ω over the alphabet Σ is a (finite) sequence of letters and is denoted by simple juxtaposition:

$\omega = x_1x_2\ldots x_n$ where $\omega_i \in \{1,\ldots,n\}$, $x_i \in \Sigma$

Example:

- abbac and bccca are two wordst on the alphabet $\{a,b,c\}$.
- 01101 is a word on the alphabet $\{0,1\}$.

Length of a word

- The number of characters (letters, digits, or other symbols) it contains. For instance, in the word sequence $\omega = x_1 x_2 \ldots x_n$, the length of the word is n, representing the total number of symbols in the sequence.

  Thus, $|abbac| = 5$ and $|ba| = 2$.

- We also define the number of occurrences of a letter d from Σ in ω; $|\omega|_d$

Example :

- let $\omega = |00011001|$ be a word in $\Sigma = \{0,1\}$ hence : $|\omega|_0 = 5$ and $|\omega|_1 = 3$.

The empty word

- The empty word is a word without symbols and therefore of length 0. This word is represented by the symbol ε ($|\varepsilon| = 0$).

Concatenation of words

- Let X be an alphabet, $x \in \Sigma*$ is a word of length m, and $y \in \Sigma*$ is a word of length n. The concatenation of x and y, denoted xy, is the word of length m+n whose first m symbols represent a word equal to x, and the last n symbols represent a word equal to y.

- More specifically, if $x=a_1a_2\ldots a_m$ and $y=b_1b_2\ldots b_n$ then

  $$xy= a_1a_2\ldots a_mb_1b_2\ldots b_n \,.$$

Concatenation of words

- Example :

Let x=01101 and y=001, then

xy=01101001 and yx=00101101.

- Concatenation is associative ((xy)z = x(yz)) but generally not commutative.
- The empty word is the neutral element for concatenation : $\varepsilon x = x \varepsilon = x$.
- Concatenation is regular on both the right and the left
  - $wu = wv \Rightarrow u = v$
  - $uw = vw \Rightarrow u = v$

- $|uv| = |u| + |v|$

Power of an alphabet :

- Let Σ un alphabet, be an alphabet, we denote by Σ^k the set of all words of a given length k over this alphabet.

- Examples :

• $\Sigma^0 = \{\varepsilon\}$ whatever the alphabet Σ.

• if $\Sigma = \{a,b\}$ then $\Sigma^1 = \{a,b\}$, $\Sigma^2 = \{aa,ab,ba,bb\}$,

$\Sigma^3 = \{aaa,aab,aba,abb,baa,bab,bba,bbb\}$, ….

Power of an alphabet :

- The set of all words over Σ is denoted by $\Sigma *$.
- For instance $\{a, b, c\}* = \{\boldsymbol{\varepsilon}$, a, b, c, aa, ab, ac, ba, bb, bc, ca, cb, cc, aaa, aab,…}.

In another way :

- $\Sigma^* = \Sigma^{\mathbf{0}} \cup \Sigma^{\mathbf{1}} \cup \Sigma^{\mathbf{2}} \cup \ldots$
- Sometimes we want to exclude the empty word from the set of words. The set of non-empty words over the alphabet X is denoted by X^+.
- $\Sigma^+ = \Sigma^{\mathbf{1}} \cup \Sigma^{\mathbf{2}} \cup \Sigma^{\mathbf{3}} \cup \ldots$
- $\Sigma^* = \Sigma^+ \cup \{\boldsymbol{\varepsilon}\}$

Miror of a word

- Let Σ be an alphabet and let $\omega \in \Sigma^*$, $|\omega| = n$ and $\omega = \omega_0 \, \omega_1 \ldots\ldots\ldots \omega_n$, $n \geq 0$
- The mirror (reverse) of ω is $\omega^R = \omega_n \omega_{n-1} \ldots \omega_1 \omega_0$

- Property:
  - $\forall$ u, v $\in \Sigma^*$: $(uv)^R = v^R u^R$

- Palindrome :

• Let Σ be an alphabet and let $\omega \in \Sigma^*$, a word is called a palindrome if : $\omega = \omega^R$

Example : $\omega = 0110 \rightarrow \omega^R = 0110 \rightarrow \omega = \omega^R$

Factors

- We call a left factor of ω a word u such that $uv = \omega$.
- We call a right factor a word v such that $uv = \omega$.
- We call a factor of w a word u such that there exist v and v' such that $vuv' = \omega$.

Example:

- "bon" is a left factor of the word "bonjour".
- "jour" is a right factor of the word "bonjour".
- " jo" is a factor of the word "bonjour".

Languages

Langages

We call a **language over Σ** any set of words over **Σ**

Definition :

- A **(formal) language** is any subset L of Σ^*, that is $L \subset \Sigma^*$.

Examples :

- $L1 = \Sigma^*$, $L2 = \emptyset$, $L3 = \{\varepsilon\}$, $L4 = \{ \omega \in \Sigma^*, \ = \omega_1 ab \omega_2\}$
- A language can be finite or infinite

Slet $\Sigma = \{0,1\}$

- $L1 = \{\omega \in \Sigma^*, \omega 1 \equiv [3]\}$, L1 is infinite.
- $L2 = \{\omega \in \Sigma^*, |\omega| < 5\}$, L2 est finite

Languages

- Note :

Among languages, it is important to distinguish:

- The set $\emptyset$ (the empty set, which contains no words).
- The language $\{\varepsilon\}$ (the language that contains only the empty word as its sole element).

Notes :

- A **finite language** is a language that contains a finite number of words.
- The **empty language** contains no words.
- A language is said to be **proper** if it does not contain the empty word.
- A language is **infinite** if it is neither empty nor finite.
- Some infinite languages (**semi-decidable languages**) can be described by a set of rules called a **formal grammar**. There are other infinite languages for which no description method exists; these are called **undecidable languages**.

Operations on languages

- Let X be an alphabet, a certain number of operations can be performed on languages:
- **union** : $\boldsymbol{L1} \cup \boldsymbol{L2} = \omega \in \boldsymbol{X*} \mid \omega \in \boldsymbol{L1}$ or $\omega \in \boldsymbol{L2}$
- **intersection** : $\boldsymbol{L1} \cap \boldsymbol{L2} = \omega \in \boldsymbol{X*} \mid \omega \in \boldsymbol{L1}$ and $\omega \in \boldsymbol{L}$
- **complement** with respect to X* : $\boldsymbol{L} = \omega \in \boldsymbol{X*} \mid \omega \notin \boldsymbol{L}$
- **difference** : $\boldsymbol{L1} - \boldsymbol{L2} = \boldsymbol{L1} \cap \boldsymbol{L2} = \omega \mid \omega \in \boldsymbol{X*} \mid \omega \in \boldsymbol{L1}$ and $\omega \notin \boldsymbol{L2}$
- **concatenation** : $\boldsymbol{L1}.\boldsymbol{L2} = \boldsymbol{u}.\boldsymbol{v} \mid \boldsymbol{u} \in \boldsymbol{L1}\ \boldsymbol{et}\ \boldsymbol{v} \in \boldsymbol{L2}$

Operations on languages

- Power of a language: $\boldsymbol{L^2 = L.L\ et}\ \forall \boldsymbol{n} \in \mathbb{N}$,

$\boldsymbol{L^{n+1} = L^n.\ avec\ L^0 = \varepsilon}$

- **(The transition to) the Kleene Star**
- The language L∗ (Kleene star of L) is defined by:

– $\boldsymbol{L^* = L^0 \cup L \cup L^2 \ldots \cup L^n \ldots =}$

$$\{\boldsymbol{u} \mid \exists \boldsymbol{n} \in \mathbb{N}, \boldsymbol{u_1}, \ldots, \boldsymbol{u_n} \in \boldsymbol{L\ tel\ que\ u = u_1 \ldots u_n}\}$$

- The plus operation : $\boldsymbol{L^+ = L^*.L = L \cup L^2 \ldots \cup L^n \ldots}$

Example

$$L_1 = \{\varepsilon, aa\},\ L_2 = \{a^i b^j / i, j \geqslant 0\} \text{ et } L_3 = \{ab, b\}.$$

$$L_1.L_2, L_1.L_3, L_1 \cup L_2, L_2 \cap L_3, L_1^{10}, L_1^*, L_1^+, L_2^R.$$

Solutions :

- $L_1.L_2 = L_2\,;$
- $L_1.L_3 = \{ab, b, aaab, aab\}\,;$
- $L_1 \cup L_2 = L_2\,;$
- $L_2 \cap L_3 = L_3\,;$
- $L_1^{10} = \{a^{2n} / 10 \geqslant n \geqslant 0\}\,;$
- $L_1^* = L_1^+ = \{a^{2n} n \geqslant 0\}\,;$
- $L_2^R = \{b^i a^j / i, j \geqslant 0\}.$

Examples of languages

$$\Sigma = \{a\} \qquad L_1 = \{\varepsilon, a, aa, aaa, \ldots\}$$

$$\Sigma = \{a, b\} \qquad L_2 = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \ldots\}$$

$$\Sigma = \{a, b\} \qquad L_2 = \{\varepsilon, ab, aabb, aaabbb, aaaabbbb, \ldots\}$$

$$\Sigma = \{a, b\} \qquad L_3 = \{\varepsilon, aa, bb, aaaa, abba, baab, bbbb, \ldots\}$$

$$\Sigma = \{a, b, c\} \qquad L_4 = \{\varepsilon, abc, aabbcc, aaabbbccc, \ldots\}$$

Description of languages

- Description in natural language : .
- Language L1 over the alphabet {0,1}: set of words whose interpretation as integers are multiples of three. It includes, for example, the word: 1001 but not 1000.
- Language L2 over the alphabet {a,b}: formed of all palindrome words. Thus, language L2 contains abbabba but not abbabab.
- L1 over the alphabet {0,1}: set of words whose interpretation as integers are multiples of three.

Description of language

- **Enumerative descriptions :**

- They are clearly used for finite languages, but also for certain infinite languages such as: :
- L3 = $\{a^n b^n \ / \ n \geq 1\}$: containing all words formed exactly of a sequence of n occurrences of the letter a followed by a sequence containing the same number of occurrences n of the letter b.

Description of languages

- **Definition by expression**:

• for example, the expression ab*cabc represents the language L_4, whose words start with an occurrence of the letter a, followed by any number (possibly zero) of occurrences of the letter b, followed by the right factor cabc

Description of languages

- **Generative mechanisms:**

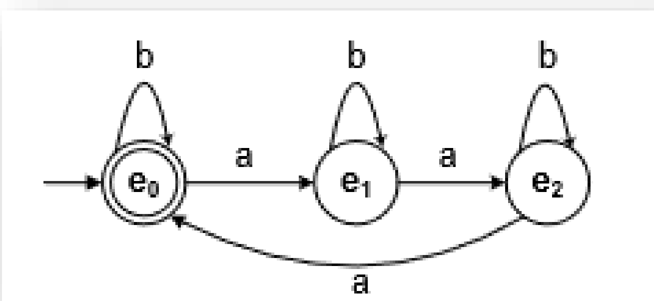
• called grammar or rewriting systems: they define a mechanism for generating words in the form of inductive construction rules.

• The language L3 produced by the rules of the following grammar: $S \longrightarrow aSb \;\; | \;\; S \longrightarrow ab$

Description of language

Recognition mechanisms:

- also called automata or machines: they allow determining whether a word belongs to the considered language or not
- Example :



$\# S_0 a \rightarrow \# a S_0$

$\# S_0 b \rightarrow \# b S_0$

$a S_0 a \rightarrow a a S_0$

$a S_0 b \rightarrow S_0$

$b S_0 a \rightarrow S_0$

$b S_0 b \rightarrow b b S_0$

$\# S_0 \rightarrow \# S_f$

Tank you

Any Questions?