

Teoria dei Linguaggi

Indice

1. Lezione 01 [26/02]	3
1.1. Cosa faremo	3
1.2. Storia	3
1.3. Ripasso	3
1.4. Gerarchia di Chomsky	4

1. Lezione 01 [26/02]

1.1. Cosa faremo

In questo corso studieremo dei sistemi formali che possiamo quindi descrivere a livello matematico. Questi sistemi descrivono dei linguaggi. Ci chiediamo giustamente cosa sono in grado di fare questi sistemi, ovvero cosa sono in grado di descrivere in termini di linguaggi.

Ci occuperemo anche delle risorse utilizzate dal sistema o delle risorse necessarie per descrivere il linguaggio. Per le prime citate, ci occuperemo del tempo come numero di mosse eseguite da una macchina riconoscitrice oppure del numero di stati per descrivere, ad esempio, una macchina a stati finiti oppure dello spazio utilizzato da una macchina di Turing. Queste ultime due questioni rientrano più nella complessità descrittiva di una macchina.

1.2. Storia

Un **linguaggio** è uno strumento di comunicazione usato da membri di una stessa comunità, ed è composto da due elementi:

- **sintassi**: insieme di simboli (o parole) che devono essere combinati/e con una serie di regole;
- **semantica**: associazione frase-significato.

Per i linguaggi naturali è difficile dare delle regole sintattiche: vista questa difficoltà, nel 1956 **Noam Chomsky** introduce il concetto di **grammatiche formali**, che si servono di regole matematiche per la definizione della sintassi di un linguaggio.

Il primo utilizzo dei linguaggi risale agli stessi anni con il **compilatore Fortran**. Anche se ci hanno messo l'equivalente di 18 anni/uomo, questa è la prima applicazione dei linguaggi formali. Con l'avvento, negli anni successivi, dei linguaggi Algol, quindi linguaggi con strutture di controllo, la teoria dei linguaggi formali è diventata sempre più importante.

Oggi la teoria dei linguaggi formali sono usati nei compilatori di compilatori, dei tool usati per generare dei compilatori per un dato linguaggio fornendo la descrizione di quest'ultimo.

1.3. Ripasso

Un **alfabeto** è un insieme non vuoto e finito di simboli, di solito indicato con Σ o Γ .

Una **stringa** x (o **parola**) è una sequenza finita $x = a_1 \dots a_n$ di simboli appartenenti a Σ .

Data una parola w , possiamo definire:

- $|w|$ numero di caratteri di w ;
- $|w|_a$ numero di occorrenze della lettera $a \in \Sigma$ in w .

Una parola molto importante è la **parola vuota** ε o λ , che, come dice il nome, ha simboli, ovvero $|\varepsilon| = |\lambda| = 0$ (ogni tanto è Λ).

L'insieme di tutte le possibili parole su Σ è detto Σ^* , ed è un insieme infinito.

Un'importante operazione sulle parole è la **concatenazione** (o prodotto), ovvero se $x, y \in \Sigma^*$ allora la concatenazione w è la parola $w = xy$.

Questo operatore di concatenazione:

- non è commutativo, infatti $w_1 = xy \neq yz = w_2$ in generale;
- è associativo, infatti $(xy)z = x(yz)$.

La struttura $(\Sigma^*, \cdot, \varepsilon)$ è un **monoide** libero generato da Σ .

Vediamo ora alcune proprietà delle parole:

- **prefisso**: x si dice prefisso di w se esiste $y \in \Sigma^*$ tale che $xy = w$;
 - ▶ **prefisso proprio** se $y \neq \varepsilon$;
 - ▶ **prefisso non banale** se $x \neq \varepsilon$;
 - ▶ il numero di prefissi è uguale a $|w| + 1$.
- **suffisso**: y si dice suffisso di w se esiste $x \in \Sigma^*$ tale che $xy = w$;
 - ▶ **suffisso proprio** se $x \neq \varepsilon$;
 - ▶ **suffisso non banale** se $y \neq \varepsilon$;
 - ▶ il numero di suffissi è uguale a $|w| + 1$.
- **fattore**: y si dice fattore di w se esistono $x, z \in \Sigma^*$ tali che $xyz = w$;
 - ▶ il numero di fattori è al massimo $\frac{|w|(|w|+1)}{2} + 1$, visti i dopponi.
- **sottosequenza**: x si dice sottosequenza di w se x è ottenuta eliminando 0 o più caratteri da w ; in poche parole, x si ottiene da w scegliendo dei simboli IN ORDINE; non devono essere caratteri contigui, basta che una volta scelti i caratteri essi siano mantenuti nell'ordine di apparizione della stringa iniziale;
 - ▶ un fattore è una sottosequenza contigua.

Un **linguaggio** L definito su un alfabeto Σ è un qualunque sottoinsieme di Σ^* .

1.4. Gerarchia di Chomsky

Vogliamo rappresentare in maniera finita un oggetto infinito come un linguaggio.

Abbiamo a nostra disposizione due modelli molto potenti:

- **generativo**: date delle regole, si parte da un certo punto e si generano tutte le parole di quel linguaggio con le regole date; parleremo di questi modelli tramite le grammatiche;
- **ricognoscitivo**: si usano dei modelli di calcolo che prendono in input una parola e dicono se appartiene o meno al linguaggio.

Considerando il linguaggio sull'alfabeto $\{(,)\}$ delle parole ben bilanciate, proviamo a dare due modelli:

- **generativo**: a partire da una sorgente S devo applicare delle regole per derivare tutte le parole appartenenti a questo linguaggio;
 - ▶ la parola vuota ε è ben bilanciata;
 - ▶ se x è ben bilanciata, allora anche (x) è ben bilanciata;
 - ▶ se x, y sono ben bilanciate, allora anche xy è ben bilanciata.
- **ricognoscitivo**: abbiamo una black-box che prende una parola e ci dice se appartiene o meno al linguaggio (in realtà potrebbe non terminare mai la sua esecuzione);
 - ▶ $\#(= \#)$;
 - ▶ per ogni prefisso, $\#(\geq \#)$.