Lezione 13 – la gerarchia di Chomsky

Lezione del 02/04/2025

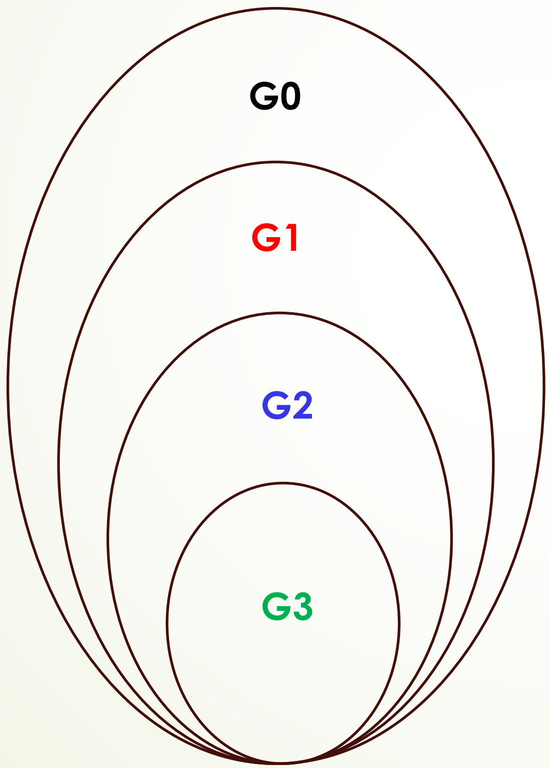
La gerarchia di Chomsky

- Le grammatiche formali vennero formalizzate da Noam Chomsky negli anni '50 del XX secolo
  - al fine di individuare uno strumento che permettesse di formalizzare le proprietà sintattiche dei linguaggi naturali
  - e raggiungendo solo molto parzialmente il suo obiettivo

- Chomsky classificò le grammatiche in quattro *tipi*
  - grammatiche di ***tipo 0***, di ***tipo 1***, di ***tipo 2*** e di ***tipo 3***
- introducendo vincoli via via più restrittivi
  - in modo tale che i vincoli delle grammatiche di tipo i sono rispettati anche dalle grammatiche di tipo i+1
    - con i = 0, 1, 2
  - ossia, le grammatiche di tipo 3 sono anche di tipo 2, che sono anche di tipo 1, che sono anche di tipo 0
- la sequenza dei quattro tipi di grammatica è conosciuta ora come ***gerarchia di Chomsky***

La gerarchia di Chomsky

- La seguente illustrazione riassume graficamente la gerarchia di Chomsky



G0 = insieme delle grammatiche di tipo 0 (o, equivalentemente, dei linguaggi generati da grammatiche di tipo 0)
G1 = insieme delle grammatiche di tipo 1 (o dei linguaggi generati da grammatiche di tipo 1)
G2 = insieme delle grammatiche di tipo 2 (o dei linguaggi generati da grammatiche di tipo 2)
G3 = insieme delle grammatiche di tipo 3 (o dei linguaggi generati da grammatiche di tipo 3)

G3 ⊆ G2 ⊆ G1 ⊆ G0

La gerarchia di Chomsky

- L'insieme delle **Grammatiche di tipo 0** (anche dette *grammatiche illimitate*) include tutte le grammatiche formali
  - ossia, qualunque grammatica formale è una grammatica di tipo 0
- Le **Grammatiche di tipo-1**, anche dette **grammatiche dipendenti dal contesto** (**context-sensitive**), che generano i linguaggi context-sensitive hanno soltanto produzioni in cui *la lunghezza della parte destra è maggiore o uguale alla lunghezza della parte sinistra*
  - ossia, produzioni che riducano la lunghezza delle parole non sono ammesse
  - ossia, una produzione del tipo aBA → c non è ammessa in una grammatica di **tipo 1**
  - ossia, una grammatica che contiene una produzione del tipo aBA → c è necessariamente soltanto una grammatica di **tipo 0**
  - invece, una prouzione del tipo aBA → caC è ammessa in una grammatica di **tipo 1**

La gerarchia di Chomsky

- Le **Grammatiche di tipo-2**, anche dette **grammatiche libere dal contesto** (**context-free**), che generano i linguaggi context-free possiedono solo produzioni la cui parte sinistra consiste solamente di un carattere non terminale
  - ossia, le produzioni di una grammatica di tipo 2 hanno tutte la forma A → α con A $\in V_N$ e $\alpha \in (V_T \cup V_N)^*$
  - così una produzione del tipo aB → cd non è ammessa in una grammatica di **tipo 2**
  - ossia, una grammatica che contiene una produzione del tipo aB → cd è necessariamente una grammatica almeno di **tipo 1**
- Osserviamo che le produzioni di una grammatica di **tipo 2** soddisfano i vincoli imposti alle grammatiche di tipo 1
  - ossia, ogni grammatica di **tipo 2** è anche una grammatica di **tipo 1** (e, neanche a dirlo, una grammatica di **tipo 0**)

La gerarchia di Chomsky

- Le **Grammatiche di tipo-3**, anche dette **grammatiche regolari**, che generano i *linguaggi regolari*, dispongono solo di produzioni la cui parte sinistra consiste di un singolo carattere non terminale e la cui parte destra consiste di un singolo simbolo terminale, che *può* essere seguito (o preceduto, ma non entrambe le forme nella stessa grammatica) da un singolo carattere non terminale
  - ossia, le produzioni di una grammatica di **tipo 3** hanno tutte la forma A → a oppure A → aB con A, B ∈ V_N e a ∈ V_T
  - o, in alternativa, la forma A → a oppure A → Ba con A, B ∈ V_N e a ∈ V_T
  - ma se una grammatica contiene le due produzioni A → aB e C → Dc (con A, B, C, D ∈ V_N e a, c ∈ V_T) allora *non* è di **tipo 3**
  - ossia, una grammatica che contiene le due produzioni A → aB e C → Dc è necessariamente una grammatica almeno di **tipo 2**
  - ancora, ogni grammatica di **tipo 3** è anche di **tipo 2** e, quindi, di **tipo 1** (e, neanche a dirlo, di **tipo 0**)

La gerarchia di Chomsky e la parola vuota

- La grammatica di tipo-0 è l'unica che può generare la parola vuota ε
  - infatti, per poter generare le parola vuota una grammatica deve necessariamente contenere una produzione della forma $\alpha \rightarrow \varepsilon$
    - dove α è una parola contenente almeno un non terminale
  - e in tale produzione la lunghezza della parte sinistra (≥ 1) è maggiore della lunghezza della parte destra (0)
  - ossia, una grammatica contenente una produzione del tipo $\alpha \rightarrow \varepsilon$ non è una grammatica di tipo 1
- Tuttavia, le grammatiche di tipo-1, 2 e 3 possono essere estese in modo tale da poter generare lo stesso linguaggio della grammatica di partenza arricchito della parola vuota:
  - aggiungiamo un nuovo non terminale S' che assumerà il ruolo di assioma (e, dunque il vecchio assioma S non sarà più assioma)
  - inseriamo la produzione $S' \rightarrow \varepsilon$
  - inseriamo la produzione $S' \rightarrow S$

La gerarchia di Chomsky e la parola vuota

- Vale, infatti, il seguente

  TEOREMA G1: sia G una grammatica di tipo t > 0 e sia G' la grammatica ottenuta

  - aggiungendo a G un nuovo non terminale S' che sarà l'assioma in G'
  - inserendo la produzione $S' \rightarrow \varepsilon$
  - inserendo la produzione $S' \rightarrow S$.

  Allora, $L(G') = L(G) \cup \{\varepsilon\}$

- Possiamo leggere il precedente teorema nel modo seguente:
  in una grammatica di tipo t > 0 la produzione $S \rightarrow \varepsilon$ è permessa, ma solo se S non appare nel lato destro di alcuna produzione.

La gerarchia di Chomsky e le ε-produzioni

- Una ε-produzione è una produzione della forma $\alpha \rightarrow \varepsilon$
  - in cui, cioè, la parte destra è la parola vuota
- Come abbiamo visto, la regola che definisce le grammatiche di tipo 1 impedisce che le grammatiche di tipo t > 0 dispongano di ε-produzioni
- Tuttavia, questa restrizione può essere rimossa per le grammatiche di tipo 2 e di tipo 3, in virtù del seguente teorema

  TEOREMA G2: sia G una grammatica di tipo t > 1 e sia G' la grammatica ottenuta aggiungendo a G un numero qualsiasi di ε-produzioni.

  Allora, L(G') = L(G) ∪ {ε}

- In virtù del precedente teorema possiamo concludere che: se rimuovendo le ε-produzioni di una grammatica G otteniamo una grammatica di tipo t > 1, allora il linguaggio generato da G è un linguaggio di tipo t aumentato della sola parola vuota
  - *e questa caratteristica si dimostrerà utile più avanti*

La gerarchia di Chomsky e le ε-produzioni

- Il teorema G2 vale soltanto per le grammatiche di tipo 2 e di tipo 3
- Quando, invece, aggiungiamo <u>arbitrarie</u> ε-produzioni a una grammatica G di tipo 1
  - ossia, produzioni del tipo $\alpha \rightarrow \varepsilon$ con $\alpha \neq S$ oppure $S \rightarrow \varepsilon$ con S che compare nella parte destra di qualche produzione
- allora è possibile che la grammatica che otteniamo sia in grado di generare più parole di quelle generate da G
- ESEMPIO: sia G la grammatica di tipo 1 definita dalle produzioni $S \rightarrow Ub$ e $U \rightarrow ab \mid S$
  - si può verificare che G genera le parole ab*bb (ossia, a seguita da almeno due b)
  - se aggiungiamo a G la produzione $S \rightarrow \varepsilon$, la grammatica risultante genera anche le parole b*b (oltre alla parola vuota)
- In effetti, vale il seguente teorema

TEOREMA G3: per ogni grammatica G di tipo 0 esiste una grammatica G' ottenuta aggiungendo a una grammatica di tipo 1 opportune ε –produzioni tale che L(G) = L(G')

Esercizi

- Classificare secondo la Gerarchia di Chomsky le grammatiche viste negli esempi precedenti

- Progettare una grammatica che generi il linguaggio

  $$L = \{ x \in \{0,1\}^*: x \text{ contiene un numero pari di } 1\}$$

ESEMPIO 4

- Questo esempio, assai più complesso dei precedenti, ci sarà molto utile più avanti
  - è da studiare bene
- Si chiede di progettare una grammatica G che generi l'insieme di parole

  $\{ xx : x \in \{a,b\}^+ \}$
  - x deve contenere almeno un carattere in {a,b}
- Esempi di parole che vogliamo generare, nel caso in cui $\Sigma = \{a,b,c\}$:
  - aa
  - baabbbaabb
  - bababbaaaababab baaaa
- ossia, sono parole che iniziano e terminano con la medesima parola x in $\{a,b\}^+$

ESEMPIO 4

- Si chiede di progettare una grammatica G che generi l'insieme di parole
  $\{ xx : x \in \{a,b\}^+ \}$
- La grammatica G = $\langle V_T, V_N, P, S \rangle$ che serve allo scopo è la seguente:
  - $V_T = \{ a,b \}$,
  - $V_N = \{S, U_a, U_b, X\}$
  - P contiene le produzioni seguenti
    - 1) $S \rightarrow a\, U_a\, S \mid b\, U_b\, S \mid X$
    - 2) $U_a\, a \rightarrow a\, U_a$, $U_a\, b \rightarrow b\, U_a$
    - 3) $U_b\, a \rightarrow a\, U_b$, $U_b\, b \rightarrow b\, U_b$
    - 4) $U_a\, X \rightarrow Xa$, $U_b\, X \rightarrow Xb$
    - 5) $U_a\, X \rightarrow a$, $U_b\, X \rightarrow b$
  - Osserviamo che G è una grammatica di tipo 0:
    - infatti, le produzioni 1)-4) soddisfano i vincoli del tipo 1
    - mentre la produzione 5) non li soddisfa

ESEMPIO 4

- Mostriamo ora che L(G) = { xx : x ∈ {a,b}⁺ }
- 1) mostriamo che { xx : x ∈ {a,b}⁺ } ⊆ L(G), ossia che, per ogni parola binaria x non vuota sull'alfabeto { a,b }, la parola xx è generabile mediante G:
  - sia x= $x_1\ x_2\ \dots\ x_n$, con x_i ∈ {a,b} per ogni i=1, ... ,n
  - applicando ripetutamente una delle produzioni S → a U_a S | b U_b S otteniamo S → $x_1\mathrm{U}_{x_1}$S $\Rightarrow_G$ $x_1\mathrm{U}_{x_1}\ x_2\mathrm{U}_{x_2}$ S $\Rightarrow_G$ ... $\Rightarrow_G$ $x_1\mathrm{U}_{x_1}\ x_2\mathrm{U}_{x_2}\ \dots\ x_n\mathrm{U}_{x_n}$ S
    - (dove U_{x_i}=0 se x_i =0, U_{x_i}=1 se x_i =1)
  - poi, applicando la produzione S → X otteniamo: $x_1\mathrm{U}_{x_1}\ x_2\mathrm{U}_{x_2}\ \dots\ x_n\mathrm{U}_{x_n}$ X
  - applicando la produzione U_{x_n}X → X x_n otteniamo $x_1\mathrm{U}_{x_1}\ x_2\mathrm{U}_{x_2}\ \dots\ x_{n-1}\mathrm{U}_{x_{n-1}}x_n$X x_n
  - applicando una delle produzioni U_ih → hU_i (con i,h ∈ {a,b}*) , otteniamo $x_1\mathrm{U}_{x_1}\ x_2\mathrm{U}_{x_2}\ \dots\ x_{n-1}x_n\mathrm{U}_{x_{n-1}}$X x_n e poi $x_1\mathrm{U}_{x_1}\ x_2\mathrm{U}_{x_2}\ \dots\ x_{n-1}\ x_n$ X $x_{n-1}x_n$
  - ... e così via fino ad ottenere $x_1\ x_2\ \dots\ x_n\ \mathrm{U}_{x_1}$X $x_2\ \dots\ x_n$
  - e infine, applicando la produzione U_{x_1}X → x_1 otteniamo $x_1\ x_2\ \dots\ x_n\ \ x_1\ x_2\ \dots\ x_n$
- [... continua la prossima lezione ...]