Proprietà di CFL

- **Pumping Lemma per CFL**: simile ai linguaggi regolari.
- **Proprietà di chiusura**: alcune delle proprietà di chiusura dei linguaggi regolari valgono anche per i CFL.
- **Proprietà di decisione**: possiamo controllare l'appartenenza e l'essere vuoto, ma, per esempio, l'equivalenza di CFL è indecidibile.

Forma normale di Chomsky

- Ogni CFL (senza ϵ) è generato da una CFG dove tutte le produzioni sono della forma

$$A \to BC, \text{ o } A \to a$$

  dove A, B, e C sono variabili, e a è un simbolo terminale. Questa è detta **forma normale di Chomsky (CNF)**.
- Esiste un algoritmo per ottenerla da una qualsiasi grammatica libera.
- Negli alberi sintattici generati da una grammatica in forma normale di Chomsky, ogni nodo ha al più due figli.

Pumping lemma per CFL

- **Informalmente**:
  In ogni stringa sufficientemente lunga di un CFL si possono trovare due sottostringhe brevi e vicine che è possibile eliminare o ripetere (insieme), ottenendo sempre stringhe del linguaggio.
- **Formalmente**:
  Sia L un CFL. Esiste una costante n tale che, se $z \in L$ e $\mid z \mid \geq n$, possiamo scrivere $z = uvwxy$ con le seguenti condizioni:
  1. $\mid vwx \mid \leq n$
  2. $vx \neq \epsilon$ (almeno una delle due è diversa da ϵ)
  3. per ogni $i \geq 0$, $uv^iwx^iy \in L$.

Dimostrazione informale

- Se la stringa w è sufficientemente lunga, l'albero sintattico che produce $w = uvwxy$ ha un simbolo non terminale che si ripete in un cammino dalla radice ad una foglia. Supponiamo $A_i = A_j$.

Dimostrazione informale

- Se la stringa w è sufficientemente lunga, l'albero sintattico che produce $w = uvwxy$ ha un simbolo non terminale che si ripete in un cammino dalla radice ad una foglia. Supponiamo $A_i = A_j$.
- Allora può essere individuato il sottoalbero con radice A_j (in rosa nella figura) e chiamato w il suo prodotto.

Dimostrazione informale

- Se la stringa w è sufficientemente lunga, l'albero sintattico che produce $w = uvwxy$ ha un simbolo non terminale che si ripete in un cammino dalla radice ad una foglia. Supponiamo $A_i = A_j$.
- Allora può essere individuato il sottoalbero con radice A_j (in rosa nella figura) e chiamato w il suo prodotto.
- Inoltre, dato il sottoalbero con radice A_i (in azzurro nella figura), chiamiamo vwx il suo prodotto.

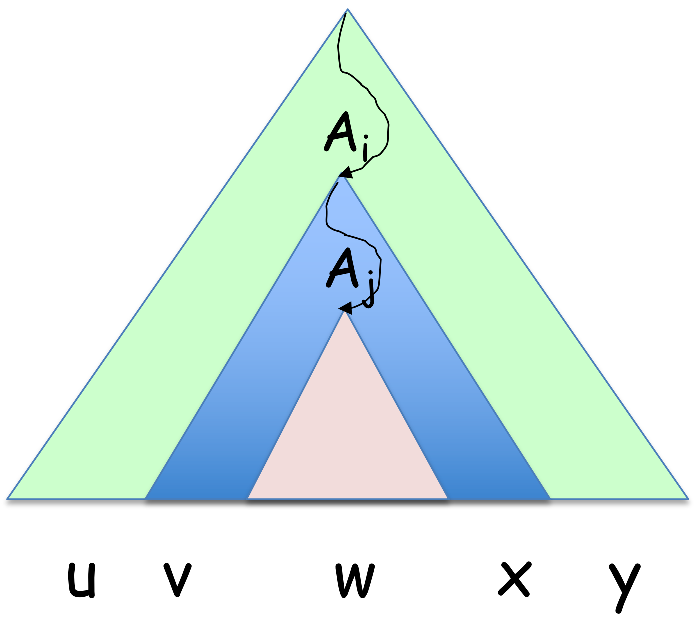
Dimostrazione informale

- Se la stringa w è sufficientemente lunga, l'albero sintattico che produce $w = uvwxy$ ha un simbolo non terminale che si ripete in un cammino dalla radice ad una foglia. Supponiamo $A_i = A_j$.
- Allora può essere individuato il sottoalbero con radice A_j (in rosa nella figura) e chiamato w il suo prodotto.
- Inoltre, dato il sottoalbero con radice A_i (in azzurro nella figura), chiamiamo vwx il suo prodotto.
- Dato che $A_i = A_j$, possiamo rimpiazzare il sottoalbero di A_i con quello di A_j, ottenendo quindi uwy $(i = 0)$, che deve ancora appartenere a L.

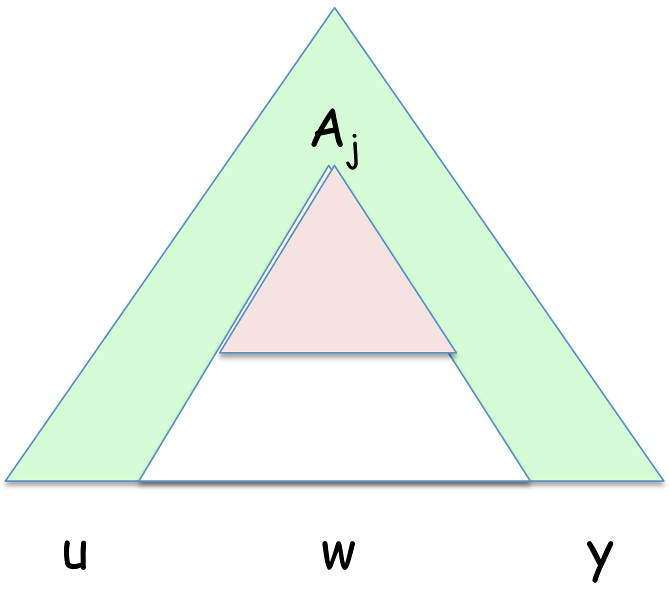
Dimostrazione informale

- Se la stringa w è sufficientemente lunga, l'albero sintattico che produce $w = uvwxy$ ha un simbolo non terminale che si ripete in un cammino dalla radice ad una foglia. Supponiamo $A_i = A_j$.
- Allora può essere individuato il sottoalbero con radice A_j (in rosa nella figura) e chiamato w il suo prodotto.
- Inoltre, dato il sottoalbero con radice A_i (in azzurro nella figura), chiamiamo vwx il suo prodotto.
- Dato che $A_i = A_j$, possiamo rimpiazzare il sottoalbero di A_i con quello di A_j, ottenendo quindi uwy $(i = 0)$, che deve ancora appartenere a L.
- Oppure possiamo rimpiazzare il sottoalbero di A_j con quello di A_i, ottenendo $uvvwxxy$ $(i = 2)$, ancora generata da L, e così via per ogni i.

Albero sintattico nel pumping lemma

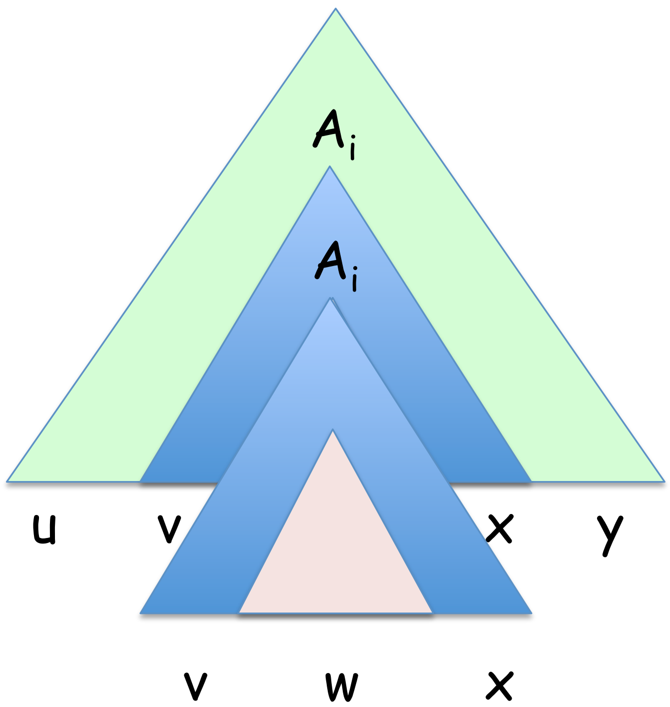


Pumping up and down



i = 0

i = 2



Proprietà

Un albero sintattico ottenuto da una CFN G il cui cammino (num. archi = num. nodi -1) più lungo sia lungo m ha un prodotto w, t.c. $|w| \leq 2^{m-1}$.

Dimostrazione

Base $n = 1$ Un albero con cammino di massima lunghezza di 1 è formato dalla radice e da un nodo foglia etichettato con un terminale. Quindi w è questo terminale e $|w| = 1 = 2^0 = 2^{m-1}$.

Ind Supponiamo che il cammino più lungo abbia lunghezza $n > 1$. La radice usa allora sicuramente una produzione di form $A \rightarrow BC$. Nessun sotto-albero con radice in B e C può avere lunghezza maggiore di $n-1$, dato che questi cammini non includono l'arco che va dalla radice a B o a C. Per ipotesi induttiva, i due sotto-alberi hanno prodotti di lunghezza massima 2^{m-2}, quindi il prodotto complessivo, concatenazione dei due, avrà lunghezza al più $2^{m-2} + 2^{m-2} = 2^{m-1}$.

Dimostrazione formale

- Sia G una grammatica libera in Forma Normale di Chomsky, t.c. $L(G) = L \setminus \{\epsilon\}$, e m il numero dei non terminali in G. Scegliamo n come 2^m e supponiamo che z sia lunga almeno n.

Dimostrazione formale

- Sia G una grammatica libera in Forma Normale di Chomsky, t.c. $L(G) = L \setminus \{\epsilon\}$, e m il numero dei non terminali in G. Scegliamo n come 2^m e supponiamo che z sia lunga almeno n.
- Un albero sintattico il cui cammino più lungo sia lungo m, o meno, ha un prodotto di lunghezza $2^{m-1} = n/2$, o meno.

Dimostrazione formale

- Sia G una grammatica libera in Forma Normale di Chomsky, t.c. $L(G) = L \setminus \{\epsilon\}$, e m il numero dei non terminali in G. Scegliamo n come 2^m e supponiamo che z sia lunga almeno n.
- Un albero sintattico il cui cammino più lungo sia lungo m, o meno, ha un prodotto di lunghezza $2^{m-1} = n/2$, o meno.
- Un albero siffatto non può avere z come prodotto, in quanto troppo lunga. Quindi ogni albero con prodotto z deve avere un cammino di lunghezza almeno $k \geq m + 1$.

Dimostrazione formale

- Sia G una grammatica libera in Forma Normale di Chomsky, t.c. $L(G) = L \setminus \{\epsilon\}$, e m il numero dei non terminali in G. Scegliamo n come 2^m e supponiamo che z sia lunga almeno n.
- Un albero sintattico il cui cammino più lungo sia lungo m, o meno, ha un prodotto di lunghezza $2^{m-1} = n/2$, o meno.
- Un albero siffatto non può avere z come prodotto, in quanto troppo lunga. Quindi ogni albero con prodotto z deve avere un cammino di lunghezza almeno $k \geq m + 1$.
- Essendoci non più di m non terminali, almeno due simboli nel cammino lungo più di $m + 1$ devono ripetersi. Supponiamo $A_i = A_j$ (where $k - m \leq i < j < k$).

Dimostrazione formale

- Sia G una grammatica libera in Forma Normale di Chomsky, t.c. $L(G) = L \setminus \{\epsilon\}$, e m il numero dei non terminali in G. Scegliamo n come 2^m e supponiamo che z sia lunga almeno n.
- Un albero sintattico il cui cammino più lungo sia lungo m, o meno, ha un prodotto di lunghezza $2^{m-1} = n/2$, o meno.
- Un albero siffatto non può avere z come prodotto, in quanto troppo lunga. Quindi ogni albero con prodotto z deve avere un cammino di lunghezza almeno $k \geq m+1$.
- Essendoci non più di m non terminali, almeno due simboli nel cammino lungo più di $m+1$ devono ripetersi. Supponiamo $A_i = A_j$ (where $k - m \leq i < j < k$).
- Allora possiamo "pompare" l'albero come uv^0wx^0y o come uv^2wx^2y, e in generale come uv^iwx^iy, $i \geq 0$.

Dimostrazione formale

- Sia G una grammatica libera in Forma Normale di Chomsky, t.c. $L(G) = L \setminus \{\epsilon\}$, e m il numero dei non terminali in G. Scegliamo n come 2^m e supponiamo che z sia lunga almeno n.
- Un albero sintattico il cui cammino più lungo sia lungo m, o meno, ha un prodotto di lunghezza $2^{m-1} = n/2$, o meno.
- Un albero siffatto non può avere z come prodotto, in quanto troppo lunga. Quindi ogni albero con prodotto z deve avere un cammino di lunghezza almeno $k \geq m+1$.
- Essendoci non più di m non terminali, almeno due simboli nel cammino lungo più di $m+1$ devono ripetersi. Supponiamo $A_i = A_j$ (where $k - m \leq i < j < k$).
- Allora possiamo "pompare" l'albero come uv^0wx^0y o come uv^2wx^2y, e in generale come uv^iwx^iy, $i \geq 0$.
- Dato che il cammino più lungo nel sottoalbero A_i è al più $m+1$, abbiamo che $|vwx| \leq 2^m = n$.

Esempio

- Consideriamo $L = \{0^n 1^n 2^n \mid n \geq 1\}$.
- Dato un n generico, scegliamo $z = 0^n 1^n 2^n$.
- Comunque noi spezziamo z in $uvwxy$, con $\mid vwx \mid \leq n$ e v e x non entrambi vuote, vwx non può contenere sia 0 che 2 perché l'ultimo 0 e il primo 2 sono lontani n+1 posti.
- Ci sono i seguenti casi:
  - vwx non contiene 2. Allora vx ha solo 0 e 1. Quindi uwy, che dovrebbe essere in L, ha n 2, ma meno di n 0 o 1.
  - vwx non contiene 0. Analogo.

Esempi

- I CFL non sanno abbinare coppie con lo stesso numero di simboli, se le coppie sono intrecciate.
  - Esempio: $L = \{0^i 1^j 2^i 3^j \mid i, j \geq 1\}$.
  - Dato n, scegliamo $z = 0^n 1^n 2^n 3^n$. Quindi vwx contiene un solo simbolo o due simboli. In ogni caso, le stringhe generate non sono in L.
- I CFL non sanno abbinare due stringhe di lunghezza arbitraria, se sono su un alfabeto di più di un simbolo.
  - Esempio: $L = \{ww \mid w \in \{0, 1\}^*\}$.
  - Dato n, scegliamo $z = 0^n 1^n 0^n 1^n$. Comunque la scomponiamo, non otteniamo stringhe di L.

Proprietà di chiusura dei CFL

Teorema 7.24: I CFL sono chiusi sotto

- unione,
- concatenazione,
- chiusura di Kleene e chiusura positiva $+$

Basta mettere insieme le grammatiche:

- per l'unione: $S \rightarrow A \mid B$
- per la concatenazione: $S \rightarrow AB$
- per la chiusura di Kleene: $S \rightarrow SA \mid \epsilon$
- per la chiusura positiva: $S \rightarrow SA \mid A$

Chiusura rispetto all'inversione

Teorema: Se L e CF, allora lo è anche anche L^R.
Prova: Supponiamo che L sia generato da $G = (V, T, P, S)$.
Costruiamo $G^R = (V, T, P^R, S)$, dove

$$P^R = \{A \to \alpha^R : A \to \alpha \in P\}$$

Si mostra per induzione sulla lunghezza delle derivazioni in G e in G^R che $(L(G))^R = L(G^R)$.

I CFL non sono chiusi sotto l'intersezione

Sia $L_1 = \{0^n 1^n 2^i : n \geq 1, i \geq 1\}$. Allora L_1 è libero da contesto, con grammatica

$$S \rightarrow AB$$
$$A \rightarrow 0A1 | 01$$
$$B \rightarrow 2B | 2$$

Inoltre, $L_2 = \{0^i 1^n 2^n : n \geq 1, i \geq 1\}$ è libero da contesto, con grammatica

$$S \rightarrow AB$$
$$A \rightarrow 0A | 0$$
$$B \rightarrow 1B2 | 12$$

Invece, $L_1 \cap L_2 = \{0^n 1^n 2^n : n \geq 1\}$ non è CF.

Intersezione tra CFL e linguaggi regolari

Si può dimostrare il seguente teorema.
Teorema 7.27: Se L è CF, e R è regolare, allora $L \cap R$ è CF.

Teorema 7.29: Siano L, L_1, L_2 CFL e R regolare. Allora

1. $L \setminus R$ è CF
2. $\bar{L}$ non è necessariamente CF
3. $L_1 \setminus L_2$ non è necessariamente CF

Prova:

1. $\bar{R}$ è regolare, $L \cap \bar{R}$ è regolare, e $L \cap \bar{R} = L \setminus R$.
2. Se $\bar{L}$ fosse sempre CF, seguirebbe che
$$L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$$
sarebbe sempre CF.
3. Notare che Σ^* è CF, quindi se $L_1 \setminus L_2$ fosse sempre CF, allora lo sarebbe sempre anche $\Sigma^* \setminus L = \bar{L}$.

Problemi indecidibili per linguaggi liberi da contesto

I seguenti problemi sono indecidibili (cioè non esiste nessun algoritmo che possa risolverli):

1. Data G, è ambigua?
2. È un dato CFL inerentemente ambiguo?
3. È l'intersezione di due CFL vuota?
4. Dati due CFL, sono uguali?
5. Dato un CFL, è uguale a Σ^*?

Alcuni esercizi di riepilogo

1. Dimostrare che $L = \{0^n \ n = k^2, k > 0\}$ non è regolare.
2. Applicando il Pumping Lemma a un linguaggio regolare "vince l'avversario" e non si può finire la dimostrazione. Dove fallisce nei casi $L = \emptyset$ e $L = \{00, 11\}$.
3. Dimostrare che $L = \{a^i b^j c^k | i < j < k\}$ non è libero.

Alcuni esercizi di riepilogo

Applicando il Pumping Lemma a un linguaggio regolare “vince l’avversario” e non si può finire la dimostrazione. Dove fallisce nei casi $L = \emptyset$ e $L = \{00, 11\}$.

- Non possiamo scegliere w dalla lingua vuota.
- Se l’avversario sceglie n $= 3$, non è possibile scegliere una w di lunghezza almeno pari a n.