Alfabeti, stringhe, linguaggi

Alfabeto

Un **alfabeto** e' un **insieme finito** di elementi, detti **simboli** o **caratteri**.

Esempi:

- $\{a, b, c\}$
- $\{0, 1\}$
- $\{\alpha, \beta, \gamma, \delta\}$

La **cardinalita'** di un alfabeto e' il numero di simboli dell'alfabeto.

Se Σ denota un alfabeto, $|\Sigma|$ denota la sua cardinalita'.

Esempi:

- $|\{a, b, c\}| = 3$
- $|\{0, 1\}| = 2$
- $|\{\alpha, \beta, \gamma, \delta\}| = 4$

Stringa su un alfabeto I

Una **stringa** o **parola** su un alfabeto e' una sequenza (o lista) di simboli appartenenti all'alfabeto.

Esempi:

- aabb, cac, cba, abba sono stringhe sull'alfabeto {a, b, c}
- un numero scritto in binario e' una stringa sull'alfabeto {0, 1}

Due parole che differiscono solo per l'ordine dei simboli sono **diverse**: aabb e abba sono due parole diverse.

La stringa vuota, denotata dal simbolo "ε" , e' la stringa che non contiene nessun simbolo.

Stringa su un alfabeto II

La **lunghezza** di una stringa e' il numero dei suoi caratteri
Se x denota una stringa, $|x|$ denota la sua lunghezza.

Esempi:

$|aabb| = 4$
$|cac| = 3$
$|101011| = 6$

Con ε si denota la *stringa vuota*. La lunghezza della stringa vuota e' 0: $|\varepsilon| = 0$

Due parole sono **uguali** solo se i loro caratteri letti ordinatamente da sinistra a destra coincidono. Formalmente:
Sia $x = a_1 . . . a_k$ e $y = b_1 . . . b_h$

$$x = y \Leftrightarrow k = h \ \& \ \forall 1 \leq i \leq k \ \ a_i = b_i$$

Due stringhe uguali hanno la stessa lunghezza, non e' vero il viceversa.

Operazioni sulle stringhe I

Il concatenamento (o concatenazione)di 2 stringhe e' la stringa formata da tutti i simboli della prima stringa seguiti da tutti quelli della seconda stringa; x.y *oppure* xy denota il concatenamento delle stringhe x e y.

se $x = a_1 \ldots a_h$ e $y = b_1 \ldots b_k$ allora $xy = a_1 \ldots a_h b_1 \ldots b_k$

Esempi:

nano.tecnologie = nanotecnologie   tele.visione = televisione

il concatenamento non e' commutativo: $x.y \neq y.x$

visione.tele = visionetele

il concatenamento e' associativo: $x.(y.z)=(x.y).z$

nano.(tecno.logie) = nano.tecnologie = nanotecnologie

(nano.tecno).logie = nanotecno.logie = nanotecnologie

il concatenamento ha un elemento neutro: $\varepsilon.x = x.\varepsilon = x$

la lunghezza della stringa concatenazione di due stringhe x e y e' la somma delle lunghezze delle stringhe x e y: $|x.y| = |x|+ |y|$

|televisione| = 11 = 4 + 7 = |tele| + |visione|

Definizioni

la stringa y e’ una **sottostringa** della stringa x se esistono delle stringhe u e v tali che x = uyv

la stringa y e’ un **prefisso** della stringa x se esiste una stringa v tale che x = yv

N.B. un prefisso e’ una sottostringa in cui u = ε

la stringa y e’ un **suffisso** della stringa x se esiste una stringa u tale che x = uy

N.B. un suffisso e’ una sottostringa in cui v = ε

una sottostringa (prefisso, suffisso) di una stringa e’ **propria** se non coincide con la stringa vuota o con la stringa stessa.

se $|x| \geq k$ indichiamo con k : x il prefisso di x di lunghezza k (inizio di lunghezza k di x)

Operazioni sulle stringhe II

Esempi:

- le sottostrighe di abbc sono {ε, a, b, c, ab, bb, bc, abb, bbc, abbc}
- le sottostrighe proprie di abbc sono {a, b, c, ab, bb, bc, abb, bbc}
- i prefissi di abbc sono {ε, a, ab, abb, abbc}
- i prefissi propri di abbc sono {a, ab, abb}
- i suffissi di abbc sono {ε, c, bc, bbc, abbc}
- i suffissi propri di abbc sono {c, bc, bbc}
- 2 : abbc = ab
- 3 : abbc = abb

Operazioni sulle stringhe III

la **riflessione** di una stringa e' la stringa ottenuta scrivendo i caratteri in ordine inverso.

x^R denota la riflessione della stringa x

$(a_1 \ldots a_h)^R = a_h \ldots a_1$

$(abbc)^R = cbba$

la riflessione gode della proprieta': $(x^R)^R = x$

la riflessione della concatenazione di due stringhe e' la concatenazione inversa delle loro riflessioni: $(xy)^R = y^R x^R$

La riflessione della stringa vuota e' la stringa vuota: $\varepsilon^R = \varepsilon$.
Vale anche $a^R = a^R$ se $a \in \Sigma$

la riflessione ha precedenza sul concatenamento: $abbc^R = abbc$

Operazioni sulle stringhe IV

la **potenza m-esima** della stringa x e' il concatenamento di x con se stessa m volte

x^m denota potenza m-esima di x

$$x^0 = \varepsilon$$
$$x^m = x^{m-1}x \qquad m > 0$$

Esempi:

- $(abbc)^3 = abbcabbcabbc$
- $(abbc)^6 = abbcabbcabbcabbcabbcabbc$
- $(aa)^2 = aaaa$

la potenza ha precedenza sul concatenamento: $abbc^3 = abbccc$

Esempi:

- $(ab)^R = ba$
- $ab^R = ab$
- $aa^2 = aaa$
- $((ab)^R)^3 = (ba)^3 = bababa$
- $((ab)^3)^R = (ababab)^R = bababa$

Definizione di ‘linguaggio’

Un **linguaggio** su un alfabeto e’ un **insieme di stringhe** su quell’alfabeto.
Le stringhe o parole di un linguaggio vengono anche chiamate **frasi**.

Esempi:

- {aabb, cac, cba, abba} e’ un linguaggio sull’alfabeto {a, b, c}
- l’insieme dei numeri scritti in binario e’ un linguaggio sull’alfabeto {0, 1}
- l’insieme delle stringhe palindrome contenenti solo i simboli a, b, c e’ un linguaggio sull’alfabeto {a, b, c}

N.B. il primo ed il terzo linguaggio hanno lo stesso alfabeto.

Dato un alfabeto quanti linguaggi si possono definire su di esso?

Linguaggi e la stringa vuota

La stringa vuota ε è una stringa come le altre e quindi può appartenere o no a un linguaggio.

Esempi:

- $\{aabb, cac, cba, \varepsilon\}$
- $\{\varepsilon\}$

sono linguaggi

N.B. poichè i linguaggi sono insiemi:

$$\{aabb, cac, cba\} \neq \{aabb, cac, cba, \varepsilon\}$$

In particolare

$$\{aabb, cac, cba\} \subset \{aabb, cac, cba, \varepsilon\}$$

Notazione insiemistica

Un linguaggio puo` essere definito mediante un descrittore di insiemi:

{w | enunciato su w}

Questa espressione va letta come “l’insieme delle parole w tali che vale l’enunciato su w scritto a destra della barra verticale”.

Alcuni esempi:

– {w | w consiste di un numero uguale di 0 e di 1}

– {w | w e` un intero binario primo}

– {w | w e` un programma C sintatticamente corretto}

Notazione insiemistica

Certe volte w viene sostituito da un'espressione con parametri secondo l'uso della teoria degli insiemi

- $\{0^n1^n \mid n \geq 1\}$

  Si legge: insieme delle stringhe formate da 0 ripetuto n volte seguito da 1 ripetuto lo stesso numero n di volte, con n maggiore o uguale a 1;

  Questo linguaggio e` formato dalle stringhe

  $$\{01, 0011, 000111, \ldots\}$$

- $\{0^n1^m \mid 0 \leq n \leq m\}$

  Si legga: insieme delle stringhe formate da un numero di 0 (eventualmente nessuno) seguiti da un numero maggiore o eguale di 1.

Cardinalita' dei Linguaggi

La **cardinalita'** di un linguaggio e' il numero delle sue stringhe
Se L denota un linguaggio, $|L|$ denota la sua cardinalita'.
Esempi:

- | {aabb, cac, cba, abba} | = 4
- | insieme dei numerali binari |= ∞

Un linguaggio e' **finito** se la sua cardinalita' e' finita: un linguaggio finito e' anche detto vocabolario.

Un linguaggio e' **infinito** se la sua cardinalita' e' infinita.

Il linguaggio **vuoto** (denotato da Φ) e' il linguaggio che non contiene alcuna stringa $|\Phi| = 0$

Definizioni e operazioni sui linguaggi I

I linguaggi sono insiemi!

L'**unione** $L_1 \cup L_2$ dei linguaggi L_1 ed L_2 e' l'insieme delle stringhe che appartengono a L_1 oppure a L_2

$$L_1 \cup L_2 = \{x \mid x \in L_1 \textit{ or } x \in L_2\}$$

L'**intersezione** $L_1 \cap L_2$ dei linguaggi L_1 ed L_2 e' l'insieme delle stringhe che appartengono sia a L_1 che a L_2

$$L_1 \cap L_2 = \{x \mid x \in L_1 \;\&\; x \in L_2\}$$

La **differenza** $L_1 - L_2$ del linguaggio L_1 meno il linguaggio L_2 e' l'insieme delle stringhe di L_1 che non appartengono a L_2

$$L_1 - L_2 = \{x \mid x \in L_1 \;\&\; x \notin L_2\}$$

Esempi:

$\{ab, abc\} \cup \{ab, aa, cb\} = \{ab, abc, aa, cb\}$

$\{ab, abc\} \cap \{ab, aa, cb\} = \{ab\}$

$\{ab, abc\} - \{ab, aa, cb\} = \{abc\}$

$\{ab, aa, cb\} - \{ab, abc\} = \{aa, cb\}$

Definizioni e operazioni sui linguaggi II

Il linguaggio L_1 e' **incluso** nel linguaggio L_2 (notazione $\mathbf{L_1 \subseteq L_2}$) se tutte le stringhe appartenenti a L_1 appartengono anche a L_2

Il linguaggio L_1 e' **propriamente incluso** nel linguaggio L_2 (notazione $\mathbf{L_1 \subset L_2}$) se tutte le stringhe di L_1 appartengono ad L_2 ed almeno una stringa di L_2 non appartiene ad L_1

Due linguaggi sono **uguali** se contengono lo stesso insieme di stringhe

$$L_1 = L_2 \Leftrightarrow L_1 \subseteq L_2 \ \& \ L_2 \subseteq L_1$$

$$L_1 = L_2 \Leftrightarrow L_1 - L_2 = L_2 - L_1 = \Phi$$

$L_1 \subseteq L_1 \cup L_2$ $\qquad$ $L_2 \subseteq L_1 \cup L_2$

$L_1 \cap L_2 \subseteq L_1$ $\qquad$ $L_1 \cap L_2 \subseteq L_2$

$L_1 - L_2 \subseteq L_1$ $\qquad$ $L_1 - L_2 \cap L_2 = \Phi$

Definizioni e operazioni sui linguaggi III

La **riflessione** del linguaggio L (notazione L^R) e' l'insieme delle stringhe riflesse di L

$$L^R = \{x \mid x = y^R \;\&\; y \in L\}$$

Esempi:

- $\{ab, abc\}^R = \{ba, cba\}$
- $L = \{a^{2n}\, b\, a^{2m+1} \mid n, m \geq 0\}$ $\qquad$ $L^R = \{a^{2m+1}\, b\, a^{2n} \mid n,m \geq 0\}$

L'insieme degli **inizi di lunghezza k** del linguaggio L (notazione k : L) e' l'insieme degli inizi di lunghezza k delle stringhe di L

$$k : L = \{k : x \mid x \in L \;\&\; |x| \geq k\}$$

Definizioni e operazioni sui linguaggi IV

Il **concatenamento** dei linguaggi L_1 ed L_2 (notazione L_1L_2) e' l'insieme ottenuto concatenando in tutti i modi possibili le stringhe di L_1 con le stringhe di L_2

$$L_1L_2 = \{x \mid x = yz \ \& \ y \in L_1 \ \& \ z \in L_2\}$$

{ab, abc}{ab, aa, cb} = {abab, abaa, abcb, abcab, abcaa, abccb}

$$L \, \Phi = \Phi = \Phi \, L$$

$$L \, \{\varepsilon\} = L = \{\varepsilon\} \, L$$

$L_1 = \{a^{2n} \mid n \geq 0\}$ $\quad$ $L_2 = \{b \, a^{2n+1} \mid n \geq 0\}$

$L_1L_2 = \{a^{2n} \, b \, a^{2m+1} \mid n, m \geq 0\}$

Definizioni e operazioni sui linguaggi V

La **potenza m-esima** del linguaggio L (notazione L^m) e' il concatenamento di L con se stesso m volte

- $L^0 = \{ \varepsilon \}$
- $L^m = L^{m-1}L \qquad\qquad m > 0$

ovvero $L^m = \underbrace{L \cdot L \cdot \ldots \cdot L}_{\text{m volte}}$

Ex: $\{ab, abc\}^2 = \{abab, ababc, abcab, abcabc\}$

$\Phi^0 = \{\varepsilon\}$ (!)

$L = \{ab, ba\}$

$L^2 = \{abab, abba, baab, baba\}$

Nota: sia $L' = \{(ab)^2, (ba)^2\} = \{abab, baba\}$. Allora $L' \subset L^2$

In generale si ha:

$$\{ x \mid x = y^m \ \& \ y \in L\} \subseteq L^m$$

Definizioni e operazioni sui linguaggi VI

La **chiusura di Kleene** (o chiusura rispetto al concatenamento) del linguaggio L (notazione L^*) e' l'unione di tutte le potenze di L

$$L^* = \bigcup_{m=0\ldots\infty} L^m = \{\varepsilon\} \cup L^1 \cup L^2 \ldots$$

$\{a, bc\}^* = \{\varepsilon, a, bc, aa, abc, bca, bcbc, \ldots\}$

Proprieta'

- $L \subseteq L^*$ (monotonicita')
- $(x \in L^*) \& (y \in L^*) \Rightarrow xy \in L^*$ (chiusura rispetto al concatenamento)
- $(L^*)^* = L^*$ (idempotenza)
- $(L^*)^R = (L^R)^*$ (commutativita' della riflessione con la chiusura di Kleene)
- $(L^m)^R = (L\ldots L)^R = L^R\ldots L^R = (L^R)^m$ … e con la potenza
- $L_1 \ (L_2 \cup L_3) = L_1 L_2 \cup L_1 L_3$ (distributività rispetto all'*unione*) (vale anche rispetto all'*intersezione*)

Definizioni e operazioni sui linguaggi VII

$\Phi^* = \{\varepsilon\}$

$\{\varepsilon\}^* = \{\varepsilon\}$

$L = \{a, ab\}$ $\quad$ $L^* = \{\varepsilon, a, ab, aa, aab, aba, aaa, aaab, \ldots\}$

$L = \{a^{2n} \mid n \geq 0\}$ $\quad$ $L^* = \{a^{2n} \mid n \geq 0\}$

La **chiusura positiva** rispetto al concatenamento del linguaggio L (notazione L^+) e' l'unione di tutte le potenze positive di L

$$L^+ = \cup_{m=1\ldots\infty} L^m = L^1 \cup L^2 \ldots$$

$\{a, bc\}^+ = \{a, bc, aa, abc, bca, bcbc, \ldots\}$

$L^* = L^+ \cup \{\varepsilon\}$ $\quad$ $L^+ = L^*L = L\,L^*$

$L^+ \subseteq L^*$ $\quad$ $\varepsilon \in L^+ \Leftrightarrow \varepsilon \in L$

Definizioni e operazioni sui linguaggi VIII

Il **linguaggio universale** o monoide libero di un alfabeto Σ e' la sua chiusura di Kleene

$$\Sigma^* = \cup_{m=0\ldots\infty} \Sigma^m : \quad \text{stringhe di lunghezza finita, ma illimitata}$$

$\{a,b\}^* = \{\varepsilon, a, b, aa, bb, ab, ba, \ldots\}$

ogni linguaggio su un alfabeto Σ e' un sottoinsieme di Σ^*

il **complemento** di un linguaggio L su un alfabeto Σ (denotato $\neg L$) è la differenza fra Σ^* ed L

$$\neg L = \Sigma^* - L$$

$\neg\{ab, ba\} = \{\varepsilon, a, b, aa, bb, aaa, \ldots\}$
$\neg\{\varepsilon, ab, aa\}_{\}} = \{a, b, aa, bb, aaa, \ldots\}$

Esempio: un linguaggio di identificatori

$\Sigma_A = \{A,B,...,Z\}$
$\Sigma_C = \{0,1, \dots, 9\}$
N.B. Σ_A e Σ_C oltre che alfabeti sono anche linguaggi le cui parole hanno lunghezza 1.

$I \subseteq (\Sigma_A \cup \Sigma_C)^*$

$I = \Sigma_A (\Sigma_A \cup \Sigma_C)^*$ [lunghezza arbitraria]

$\Sigma = \Sigma_A \cup \Sigma_C$

$I_5 = \Sigma_A (\Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \Sigma^4) = \Sigma_A (\varepsilon \cup \Sigma)^4$ (lunghezza ≤ 5)

Esempio: numerali binari

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

$B = \{1\}\Sigma^* \cup \{0\}$

Linguaggi artificiali e formali

I linguaggi “formali”, in senso lato, sono linguaggi in cui l’insieme delle stringhe che li costituiscono è definibile in modo rigorose e formale.

Esempi: - il linguaggio naturale (almeno per ora) non è un linguaggio formale.
 - Il linguaggio dei numeri binari è un linguaggio formale.

Cosa si richiede ad un linguaggio formale?
- struttura delle frasi descritta in modo chiaro e comprensibile (sintassi).
- possibilità di definire algoritmi di riconoscimento
- possibilità di associare regole per definire il significato delle frasi (semantica).

Come descrivere i linguaggi formali

La semplice notazione insiemistica *non* è sufficiente. Bisogna ricorrere a formalismi più specifici.

Due approcci, **generativo** e **riconoscitivo**

- i formalismi generativi (*grammatiche, espr. regolari*) permettono si indirizzano alla generazione delle frasi del linguaggio attraverso la descrizione della loro struttura.
- i formalismi riconoscitivi (*automi*) forniscono algoritmi per decidere se una frase appartiene o no al linguaggio

I due approcci sono tipicamente *duali ed equivalenti*: si può passare da un algoritmo di generazione ad uno di riconoscimento in modo meccanico.