Elementi di Bioinformatica

Gianluca Della Vedova

Univ. Milano–Bicocca
https://gianluca.dellavedova.org

2 ottobre 2023

Karp-Rabin

Alfabeto binario

Karp-Rabin

Alfabeto binario

- $H(S) = \sum_{i=1}^{|S|} 2^{i-1} H(S[i])$

Karp-Rabin

Alfabeto binario

- $H(S) = \sum_{i=1}^{|S|} 2^{i-1} H(S[i])$
- sliding window di ampiezza m su T

Karp-Rabin

Alfabeto binario

- $H(S) = \sum_{i=1}^{|S|} 2^{i-1} H(S[i])$
- sliding window di ampiezza m su T
- $H(T[i+1:i+m]) =$
  $= (H(T[i:i+m-1]) - T[i])/2 + 2^{m-1}T[i+m]$

Karp-Rabin

Alfabeto binario

- $H(S) = \sum_{i=1}^{|S|} 2^{i-1} H(S[i])$
- sliding window di ampiezza m su T
- $H(T[i+1:i+m]) =$
  $= \left(H(T[i:i+m-1]) - T[i]\right)/2 + 2^{m-1}T[i+m]$
- operazioni su bit

Karp-Rabin

Alfabeto binario

- $H(S) = \sum_{i=1}^{|S|} 2^{i-1} H(S[i])$
- sliding window di ampiezza m su T
- $H(T[i+1 : i+m]) =$
  $= (H(T[i : i+m-1]) - T[i]) / 2 + 2^{m-1} T[i+m]$
- operazioni su bit
- $T[i : i+m-1] = P \Leftrightarrow H(T[i : i+m-1]) = H(P)$

Karp-Rabin: problema

Numeri troppo grandi

Karp-Rabin: problema

Numeri troppo grandi

- Modello RAM: numeri $O(n + m)$

Karp-Rabin: problema

Numeri troppo grandi

- Modello RAM: numeri $O(n + m)$
- mod p

Karp-Rabin: problema

Numeri troppo grandi

- Modello RAM: numeri $O(n + m)$
- mod p
- $H(T[i+1 : i+m]) =$
  $\left((H(T[i : i+m-1]) - T[i])/2 + 2^{m-1}T[i+m]\right) \mod p$

Karp-Rabin: problema

Numeri troppo grandi

- Modello RAM: numeri $O(n + m)$
- mod p
- $H(T[i+1 : i+m]) =$
  $\left((H(T[i : i+m-1]) - T[i])/2 + 2^{m-1}T[i+m]\right) \mod p$
- **NO**

Karp-Rabin: problema

Numeri troppo grandi

- Modello RAM: numeri $O(n+m)$
- $\bmod p$
- $H(T[i+1:i+m]) =$
  $\left((H(T[i:i+m-1]) - T[i])/2 + 2^{m-1}T[i+m]\right) \mod p$
- **NO**
- $2^{m-1}T[i+m] \mod p$ calcolato iterativamente, $\mod p$ ad ogni passo

Karp-Rabin: falsi positivi

Possibili errori

Karp-Rabin: falsi positivi

Possibili errori

- Falso positivo (FP): occorrenza non vera

Karp-Rabin: falsi positivi

Possibili errori

- Falso positivo (FP): occorrenza non vera
- Falso negativo (FN): occorrenza non trovata

Karp-Rabin: falsi positivi

Possibili errori

- Falso positivo (FP): occorrenza non vera
- Falso negativo (FN): occorrenza non trovata
- $H(T[i : i + m - 1]) = H(P) \Leftrightarrow T[i : i + m - 1] = P$

Karp-Rabin: falsi positivi

Possibili errori

- Falso positivo (FP): occorrenza non vera
- Falso negativo (FN): occorrenza non trovata
- $H(T[i : i + m - 1]) = H(P) \Leftrightarrow T[i : i + m - 1] = P$
- $H(T[i : i + m - 1]) \mod p = H(P) \mod p \Leftarrow T[i : i + m - 1] = P$

Karp-Rabin: falsi positivi

Probabilità di errore

$P[\#FP \geqslant 1] \leqslant O(nm/I)$ se il numero primo p è scelto fra tutti i primi $\leqslant I$

Karp-Rabin: falsi positivi

Probabilità di errore

$P[\#FP \geqslant 1] \leqslant O(nm/I)$ se il numero primo p è scelto fra tutti i primi $\leqslant I$

Valori di I

Karp-Rabin: falsi positivi

Probabilità di errore

$P[\#FP \geqslant 1] \leqslant O(nm/I)$ se il numero primo p è scelto fra tutti i primi $\leqslant I$

Valori di I

- $I = n^2m \Rightarrow P[\#FP \geqslant 1] \leqslant 2.54/n$

Karp-Rabin: falsi positivi

Probabilità di errore

$P[\#FP \geqslant 1] \leqslant O(nm/I)$ se il numero primo p è scelto fra tutti i primi $\leqslant I$

Valori di I

- $I = n^2 m \Rightarrow P[\#FP \geqslant 1] \leqslant 2.54/n$
- $I = nm^2 \Rightarrow P[\#FP \geqslant 1] \in O(1/m)$

Karp-Rabin: falsi positivi

Probabilità di errore

$P[\#FP \geqslant 1] \leqslant O(nm/I)$ se il numero primo p è scelto fra tutti i primi $\leqslant I$

Valori di I

- $I = n^2 m \Rightarrow P[\#FP \geqslant 1] \leqslant 2.54/n$
- $I = nm^2 \Rightarrow P[\#FP \geqslant 1] \in O(1/m)$

Abbassare probabilità di errore

Scegliere k primi casuali (indipendenti senza ripetizioni), cambiare primo dopo ogni FP

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce
  - Forse non corretto

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce
  - Forse non corretto
  - Karp-Rabin

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce
  - Forse non corretto
  - Karp-Rabin
- Las Vegas:

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce
  - Forse non corretto
  - Karp-Rabin
- Las Vegas:
  - Sempre corretto

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce
  - Forse non corretto
  - Karp-Rabin
- Las Vegas:
  - Sempre corretto
  - Forse non veloce

Las Vegas vs. Monte Carlo

Classificazione algoritmi probabilistici

- Monte Carlo:
  - Sempre veloce
  - Forse non corretto
  - Karp-Rabin
- Las Vegas:
  - Sempre corretto
  - Forse non veloce
  - Quicksort con pivot random

Controllo falsi positivi

L: posizioni iniziali in T delle occorrenze

Run

sequenza $\langle l_1, \ldots, l_k \rangle$ di posizioni in L distanti al massimo $m/2$

Controllo falsi positivi

L: posizioni iniziali in T delle occorrenze

Run

sequenza $\langle l_1, \ldots, l_k \rangle$ di posizioni in L distanti al massimo $m/2$

- $d = l_2 - l_1$

Controllo falsi positivi

L: posizioni iniziali in T delle occorrenze

Run

sequenza $\langle l_1, \ldots, l_k \rangle$ di posizioni in L distanti al massimo $m/2$

- $d = l_2 - l_1$
- P semiperiodico con periodo d

Controllo falsi positivi

L: posizioni iniziali in T delle occorrenze

Run

sequenza $\langle l_1, \ldots, l_k \rangle$ di posizioni in L distanti al massimo $m/2$

- $d = l_2 - l_1$
- P semiperiodico con periodo d
- $P = \alpha\beta^{k-1}$, α suffisso di β

Controllo falsi positivi

L: posizioni iniziali in T delle occorrenze

Run

sequenza $\langle l_1, \dots, l_k \rangle$ di posizioni in L distanti al massimo $m/2$

- $d = l_2 - l_1$
- P semiperiodico con periodo d
- $P = \alpha\beta^{k-1}$, α suffisso di β
- ogni run occupa $\geqslant n$ caratteri di T

Controllo falsi positivi

L: posizioni iniziali in T delle occorrenze

Run

sequenza $\langle l_1, \ldots, l_k \rangle$ di posizioni in L distanti al massimo $m/2$

- $d = l_2 - l_1$
- P semiperiodico con periodo d
- $P = \alpha\beta^{k-1}$, α suffisso di β
- ogni run occupa $\geqslant n$ caratteri di T
- ogni carattere di T è in max 2 run

Licenza d’uso