The Uses Of Textual Data Compression In Bioinformatic's Field

Introduction

Vogliamo mostrate il fondamentale aiuto apportato dall'information thoery nel campo biologico.

In particolare le tecniche di textual data compression sono risultate fondamentali nel campo dello studio delle sequenze biologiche (classificazione, compressione, complessità, entropia).

Compression for Storage

Due metodi principali per la compressione di sequenze di DNA:

- Orizzontale: compressione di una sequenza, utilizzando informazioni presenti solo nella sequenza.
- Verticale: compressione di un set di sequenze utilizzando informatzioni presenti nell'intero set; legato al concetto di RC(relative compressibility), ovvero potrebbe risultare difficile comprimere ogni sequenza, ma un gruppo di sequenze simili potrebbero essere compresse meglio.

In questo modo studiamo le proprietà strutturali della sequenza e riduciamo la dimensione.

In generale si usa l'orizzontale per comprimere, il verticale per classificare.

Compression for Storage

Substitutioanl-statistical methods

idea di base: la stringa è suddivisa in gruppi di sottostringhe, in un gruppo si utilizzano metodi di sostituzione per comprime, nell'altro si usano metodi statistici.

Viene usata una funzione per stabilire la divisione delle sottostringhe nei due gruppi.

Substitutional: si trovano ripetizioni di sottostringhe della stringa e si sostituisco con un puntatore.

Statical: si usa un modello di compressione della stringa che tiene conto dei caratteri che sono più frequenti (Huffman).

Compression for Storage

Trasformational methods

Idea di bade: trasformazione della stringa prima della compressione.

Trasformazione di Burrows and Wheeler

Ragruppa le ripetizione di simboli che occorrono nella stringa, il che porta ad una migliore compressione.

Si crea lista ciclica di shift della stringa x; si ordina la lista; si prende la permutazione risultante.

Compression for Storage

Grammar-based methods

La stringa x viene compressa usando una grammatica context-free G(x).

In suguito la stringa viene codificata usando le regole della grammatica.

2 metodi:

1) ad x è assegnata una grammatica G(x) che definisce il linguaggio L(x); le produzioni di G sono considerate come simboli di una nuova stringa da trasmettere.

2) G(x) viene definita a priori; dato x, solo un sotto-insieme delle produzioni in G è rilevante per generare x.

Compression for Storage

Table compression

Si usano tabelle che contengono records di lunghezza fissa, i dati vengono successivamente inseriti in grossi data-base.

La tabella deve essere veloce, on-line e consistente.

E' un potente strumento nella classificazione e nel raggruppamento di dati biologici in forma tabulare.

Entropy Estimators

Valuta il livello randomico dei simboli di una stringa.

Importante per comprendere il modello e la struttura della sequenza.

Sia Σ un alfabeto di simboli, sia $p_i(n)$ la probabilità che la i-esima stringa sia in Σ^n(ordinato lessicograficamente) e sia H(S) l'entropia dell'informazione iniziale S.

Si possono utilizzare le stringhe prodotte da S per stimare $p_i(n)$, ciò porta comunque ad una sottistima di H(S) poiché all'aumentare di n, solo una piccola frazione delle possibili stringhe in $|\Sigma^n|$ saranno nella sequenza. Ciò porta ad un'approsimazione sbagliata di $p_i(n)$ e di H(S) (finite sample effect)

Entropy Estimators

Esistono tre principali metodi che danno una stima più accurata dell'entropia.

Methods based on the AEP

L'AEP è equivalente alla legge dei grandi numeri, ovvero per stringhe abbastanza lunghe $1/n\ p_i(n)=H(S)$.

Per n abbastanza grande possiamo partizionare Σ^n in due gruppi, uno con stringhe che hanno probabilità zero, e l'altro con stringhe che hanno uguale probabilita divera da zero. Bisogna quindi stimare il valore $p_i(n)$, cio può essere fatto stimando il numero N* di stringhe che hanno probabilità diversa da zero.

Entropy Estimators

Methods based on universality theorems:

Data una stringa di lunghezza maggiore ad una certa soglia fissata, ed un algortimo di compressione C, l'algoritmo convergerà al valore H(S). C può quindi essere usato per stimare il valore entropico.

Contro: molto spesso C converge troppo lentamente.

Methods based on Renyi Entropy

È una generalizzazione della funzione entropica di Shannon, ed è una buona stima della randomicità delle sequenze di DNA.

Self-indexes

Self-indexes: strutture dati analoghi ad alberi dei suffissi e arrays ma con spazio teoricamente vicino all'entropia della sequenza da idicizzare, con velocità di ricerca e capacità di ricostruire porzioni di sequenze.

Compressed suffix array (CSA) usati per:

per indicizzare sequenze del genoma umano,(con soli 2GB di spazio, ed 1GB per sequenza DNA) e per confrontarle.

Sequence alignment

E’ legato al concetto di complessità di una sequenza.

Esistono due importanti approcci al problema:

1) definire la complessità di una sequenza basandosi su un appropiato parsing su di essa, con dizionario di sotto-sequenze di altre sequenze.
2) universal similarity metric (USM)

Quest’ultima è basasa su K(x) e K(x|y) (Kolmogorov Complexity).

Kolmogorov Complexity

Conditional Kolmogorov Complexity $K(x|y)$ di sue stringhe x e y è la lunghezza del piu piccolo programma P che calcola x con input y $K(x|y)$. Rappresenta la minima informazione necessaria per generare x avendo come input y.

Kolmogorov Complexity $K(x)$ è definita come $K(x|\lambda)$ dove λ è la stringa vuota. Adesso possiamo definite USM come:

$$USM = \max\{K(x|y),K(y|x)\} / \max\{K(x), K(y)\}$$

Sfortunamente USM non è Turing-calcolabile, ci servono quindi approssimazioni.

Kolmogorov Complexity

Se x e y vengono compresse meglio insieme che separatamente allora devono avere una qualche relazione (concetto di RC).

Tre tipi di approssimazioni per USM: UDC, NC, CD.

RC e USM vengono quindi usati per calcolare similarità tra sequenze. Usati per costruire alberi di filogenesi e per classificazione strutturale ed evolutiva delle proteine.

Segmentation of Biological Sequences

Nel DNA sono presenti zone omogenee (alta presenza di nucleotidi C e G).

E' quindi utile suddividere il DNA in segmenti omogenei.

Due principali tecniche.

1)Single Nucleotide Polymorphism and Identification of Haplotype Blocks:

Un polimorfismo a singolo nucleotide (SNP) è un polimorfismo (cioè una variazione a livello di una sequenza di acidi nucleici) che si presenta tra individui della stessa specie, caratterizzata da una differenza a carico di un unico nucleotide.

SNP and Identification of Haplotype Blocks

Si definisce aplotipo una sequenza di SNPs, risulta quindi importante dividere insieme di aplotipi in blocchi, caratterizzati da SNPs vicini.

La ricerca di blocchi aplotipi richiede la partizione di un set di sequenze in blocchi, dove l'omogeneetà tra blocchi è calcolata attraverso un'opportuna funzione costo.

Esistono una moltitudine di metodi, tutti basati su una formulazione in termini di programmazione dinamica del problema.

DNA segmentation and coding regions identifications

Data una sequenza, si identificano i punti al suo interno dove è presente un cambio di omogeneità.

Si dividono le sequenze in blocchi adiacenti, i blocchi non omogenei adiacneti vengono identificati e partizionati in blocchi più piccoli, per avere una stima precisa dei punti di cambio di omogeneità.

DNA segmentation and coding regions identifications

Un altro importante problema è quello di trovare le regioni di codifica del DNA.

Come prima la sequenza viene divisa in blocchi e venogono identificati ogni cambiamento di informazione di blocchi adiacenti.

Particolarmente importanti sono i blocchi dove le informazioni aumentano sub-linearmente rispetto alla lunghezza del blocco, ciò indica la regolarità delle sequenza in input.

Grazie a queste informazioni possiamo distinguere zone di codifica da zone di non-codifica in una sequenza di DNA.

Pattern Discovery

Vogliamo identificare sotto-sequenze di una sequenza che sono "significanti" rispetto ad una forma di misura.

Una sequenza è "significante" se il numero di bit richiesti da un programma di compressione C, è minore di quello richiesto dalla codifica della massima entropia della stringa (algorithm significance AS).

Si divide la sequenza in segmenti e successivamente viene utilizzato AS per stabilire la "significanza" di ogni segmento.

Altro problema: vogliamo individuare ripetizioni di sotto-sequenze di una sequenza di DNA (es. meccaniscmo riproduzione molecolare, copia di geni) ancora una volta le tecniche di compressione ci aiutano.

Comparison and Inference of Biological Networks

Goal:

- confronto di networks biologici;
- reverse-engineering (dai dati ai genomi) di networks biologici.

Idea di base: dato un set di elementi rappresentati da nodi, si costruisce un grafo pesato, dove il peso di ogni arco è dato dal livello di correlazione dei due nodi (ovvero una forma di misura delle informazioni comuni).

Comparison and Inference of Biological Networks

Successivamente gli archi con peso minore o uguale a zero vengono rimossi per ottenere il reverse engineered network.

Bisogna valutare accuratamente la muta informazione sui nodi che deve essere dedotta da dati empirici.

Bisogna filtrare i falsi positivi in particolare:

se x interagisce con y e z, quando non c'è interazione tra y e z la mutua informazione dovrebbe rilevare un falso positivo tra y e z.

Conclusion and pratical examples

Abbiamo mostrato come l'information theory sia di fondamentale aiuto nel campo della bioinforatica ed in generale nel campo della biologia.

Di seguito riportiamo esempi pratici di implementazione di algoritmi trattati:

- Biological data compression per la compressione di sequenze DNA (Manzini and Rastero, 2005a);
- GeneCompress;
- Entropy estimators, CDNA;
- implementazione CSA per DNA;
- NCBI taxonomy database, classificazione genoma;
- classificazione sistema proteico basata su USM (Barthel *et al.*, 2008);
- ARACNE software per il reverse engineering di networks cellulari