Università degli Studi di Milano - Bicocca
Scuola di Scienze
Dipartimento di Informatica, Sistemistica e Comunicazione
Corso di laurea in Informatica

Estensione di ASGAL per read paired-end

Relatore: Prof. Della Vedova Gianluca

Correlatore: Prof. Rizzi Raffaella

Relazione della prova finale di:
Francesco Porto
Matricola 816042

Anno Accademico 2018-2019

Abstract

In questa tesi si discuterà l'estensione di ASGAL (un tool sviluppato dall' AlgoLab in grado di rilevare eventi di Alternative Splicing a partire da campioni di RNA-Seq) per il supporto alle read in formato paired-end. Dopo aver introdotto i concetti necessari per comprendere questo documento, verranno evidenziate le principali modifiche apportate ad ASGAL, ponendo l'attenzione sulle differenze tra il formato single-end e quello paired-end. Verrà poi mostrato un esempio di funzionamento. Infine saranno discussi alcuni possibili sviluppi futuri.

Contents

1 Introduzione

1.1 ASGAL

ASGAL (Alternative Splicing Graph Aligner) è un tool per l'identificazione di eventi di Alternative Splicing espressi in un campione di RNA-seq a partire da un'annotazione di un gene. ASGAL si compone di tre step:

- **Costruzione dello splicing graph**: a partire dall'annotazione di un gene, ASGAL costruisce uno splicing graph, ovvero una struttura a grafo che rappresenta tutti i trascritti noti del gene in input.
- **Allineamento Splice-Aware**: ASGAL allinea le read di RNA-Seq con lo splicing graph del gene in input. L'allineamento è Splice-Aware in quanto è necessario tenere traccia della posizione di esoni ed introni per un corretto allineamento.
- **Rilevamento degli eventi di Alternative Splicing**: gli allineamenti prodotti dallo step precedente sono analizzati per rilevare gli eventi di alternative splicing indotti dalle read del campione.

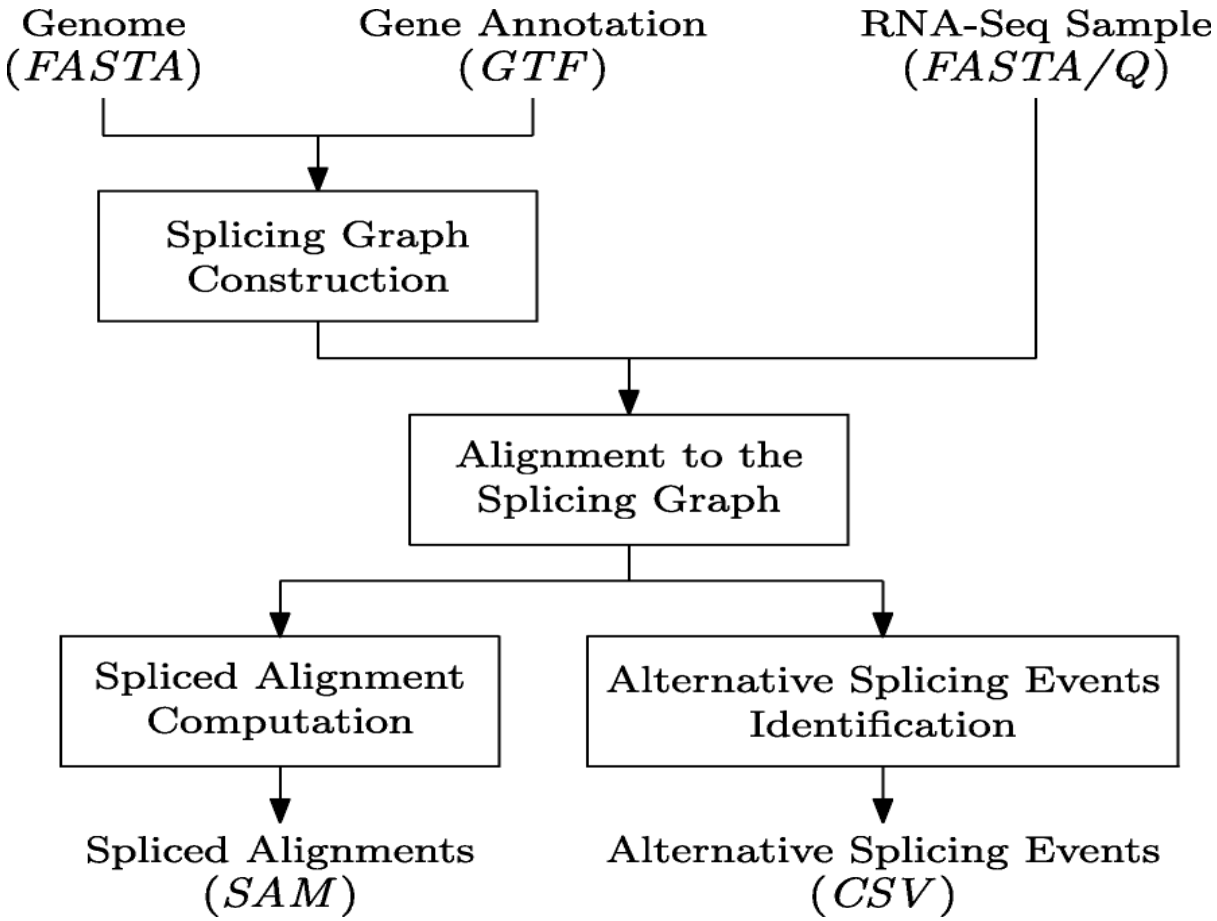

Figure 1: La pipeline di ASGAL illustrata

1.2 Alternative Splicing

L'Alternative Splicing è un metodo utilizzato dalle cellule per produrre proteine diverse dallo stesso frammento di DNA che viene utilizzato da oltre il 75% dei geni umani.
Considerando un generico locus, esso può essere diviso in esoni (parti codificanti) e introni (parti non codificanti). Durante la fase di Trascrizione gli introni vengono rimossi e la Timina viene trasformata in Uracile, ottenendo pre-RNA. A questo punto, in un normale processo di Splicing, tutti gli esoni vengono utilizzati, nell'ordine in cui appaiono nel pre-RNA, per ottenere una proteina. Nel caso di un evento di Alternative Splicing, questo non accade: alcuni esoni potrebbero infatti non essere utilizzati, o apparire in un ordine diverso.
Vengono riconosciuti 5 tipi di eventi di Alternative Splicing:

1. **Exon Skipping**: Almeno un esone non appare nel trascritto
2. **Mutually Exclusive Exons**: Almeno due esoni non compaiono mai in uno stesso trascritto
3. **Alternative 5' Donor Site**: Parte di un introne nel 5' diventa un esone
4. **Alternative 3' Acceptor Site**: Parte di un introne nel 3' diventa un esone
5. **Intron Retention**: Parte di un esono diventa un introne

ASGAL è in grado di rilevarli tutti tranne il caso 2.

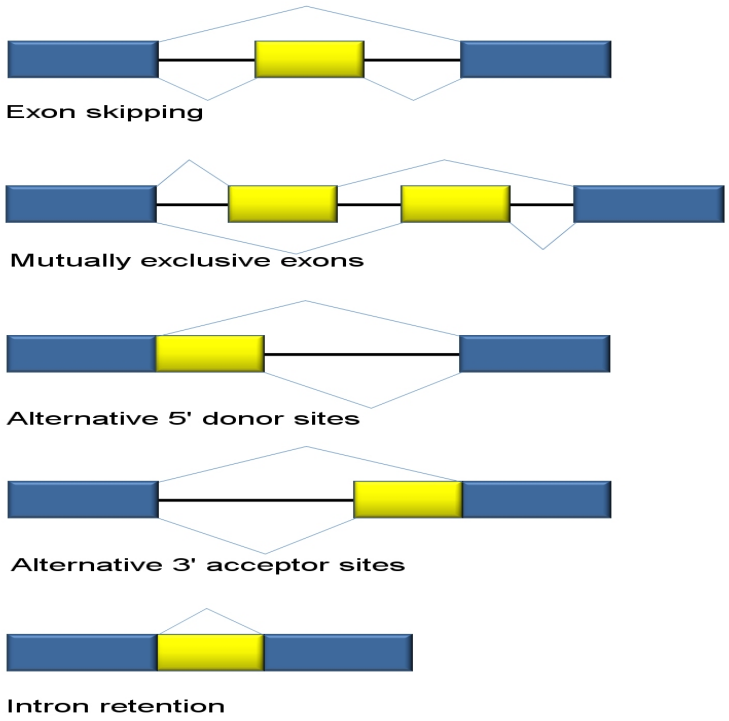

Figure 2: I diversi tipi di Alternative Splicing

1.3 Paired-End Reads

Le paired-end reads consistono nell'estrazione di due letture da un singolo frammento di DNA (generalmente le due estremità), contrariamente alle single-end reads che ne estraggono solo una. Sono prodotte da sistemi NGS, e la loro preparazione è molto semplice: una volta stabilita la grandezza della singola lettura, viene estratta la lettura sull'estremità sinistra, il campione viene girato, e viene estratta nuovamente l'estremità sinistra (ottenendo quindi l'estremità destra). Viene inoltre fornita la distanza tra le due letture, che permette di disambiguare alcuni casi di allineamento.

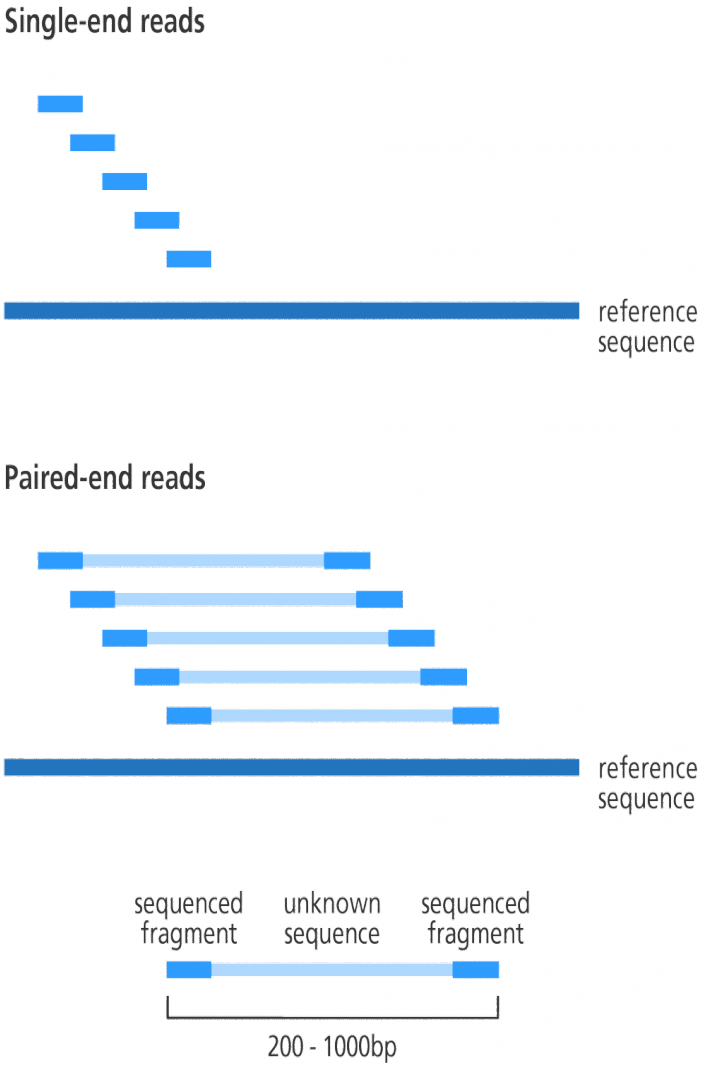

Figure 3: Esempi di read single-end e paired-end

2 Modifiche allo Splice-Aware Aligner

Lo Splice-Aware Aligner svolge due compiti:

1. Generazione dello Splicing Graph e della sua Linearizzazione
2. Allineamento delle read di RNA-Seq alla Linearizzazione dello Splicing Graph

L'allineamento avviene utilizzando il concetto di MEM (Maximum Exact Matching); l'output verrà convertito in formato SAM per permettere l'eventuale elaborazione con altri strumenti.

2.1 Allineamento di entrambe le read

Il primo problema da affrontare è ovviamente il fatto che sia ora necessario allineare due read e non una; fortunatamente si tratta solo di iterare il processo di allineamento su una coppia di read ad ogni ciclo, anziché su read singola. Verranno quindi generati due file contenti MEM anziché uno. Sarà poi compito della Formattazione SAM "fondere" i due file MEM per ottenere un SAM Completo.

2.2 Introduzione di read unmapped e "placeholder" nel formato MEM

Nei file MEM ottenuti dallo Splice-Aware Aligner vengono ora visuallizati due nuovi tipi di MEM: quelli relativi alle read unmapped e quelli relativi ai "placeholder". Il primo caso è banale, e rappresenta tutte quelle read che non hanno un matching esatto di lunghezza considerevole con il genoma dato in input. Il secondo caso è più complesso e rappresenta un insieme di read fasulle utilizzate solo come padding per avere due file MEM della stessa lunghezza: questo facilita enormemente l'elaborazione nello step successivo (la formattazione SAM). Come detto in precedenza quando si lavora con read paired-end è sempre necessario lavorare a coppie, ma non sempre ad uno stesso pair è associato lo stesso numero di allineamenti secondari: è qui che entrano in gioco i "placeholder". La loro implementazione è banale: si tengono due contatori (che rappresentano rispettivamente il numero di allineamenti relativi alla prima read e quelli relativi alla seconda read), si ottengono separatamente gli allineamenti relativi a ciascuna delle due read, e si controllano i contatori. Si prende il minore dei due e si aggiungono tanti placeholder quanto bastano per rendere uguali i contatori.

Algorithm 1 Algoritmo per l'aggiunta dei placeholder

```
1: count_aligns_1 ← 0, count_aligns_2 ← 0
2: while count_aligns_1 != count_aligns_2 do
3:     if count_aligns_1 < count_aligns_2 then
4:         addPlaceholder(file1)
5:         count_aligns_1++
6:     else
7:         addPlaceholder(file2)
8:         count_aligns_2++
9:     end if
10: end while
```

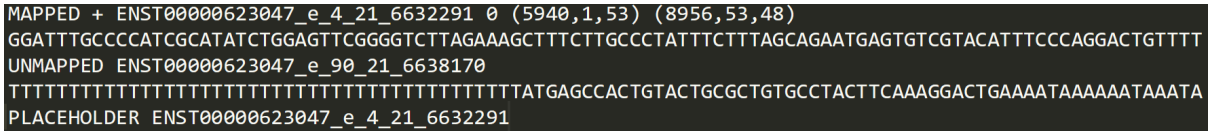

Figure 4: I diversi tipi di MEM

2.3 Supporto alle fragment library types

3 Modifiche alla Formattazione SAM

4 Modifiche alla Rilevazione di Eventi di Alternative Splicing

5 Esempio di funzionamento - gene ENSG00000280145 chr21

6 Conclusioni