Trinity Framework for De novo Transcriptome Assembly and Analysis



Assembly Required



From: http://www2.fml.tuebingen.mpg.de/raetsch/members/research/transcriptomics.html

RNA-Seq reads



Advancing RNA-Seq analysis

Brian J Haas & Michael C Zody

Nature Biotech, 2010

New methods for analyzing RNA-Seq data enable de novo reconstruction of the transcriptome.















Graph Data Structures Commonly Used For Assembly



Graph Data Structures Commonly Used For Assembly



Read Overlap Graph: Reads as nodes, overlaps as edges



Read Overlap Graph: Reads as nodes, overlaps as edges





Read Overlap Graph: Reads as nodes, overlaps as edges



Generate consensus sequence where reads overlap



Finding pairwise overlaps between *n* reads involves $\sim n^2$ comparisons.



Impractical for typical RNA-Seq data (50M reads)

No genome to align to... De novo assembly required



Want to avoid n^2 read alignments to define overlaps

Use a de Bruijn graph

Generate all substrings of length k from the reads

k-mers (k=5)

ACCGCCCACAGCGCTTCCTGCTGGTCTCTTGTTG

CGCCCTCAGCGCTTCCTCTTGTTGGTCGTAG } Reads

Generate all substrings of length k from the reads

k-mers (k=5)

ACCGC

ACCGCCCACAGCGCTTCCTGCTGGTCTCTTGTTG

CGCCCTCAGCGCTTCCTCTTGTTGGTCGTAG } Reads



Construct the de Bruijn graph



Nodes = unique k-mers



Construct the de Bruijn graph



Nodes = unique k-mers Edges = overlap by (k-1)



Construct the de Bruijn graph



Nodes = unique k-mers Edges = overlap by (k-1)



Construct the de Bruijn graph



Nodes = unique k-mers Edges = overlap by (k-1)

Generate all substrings of length k from the reads



Construct the de Bruijn graph



Nodes = unique k-mers Edges = overlap by (k-1)

Construct the de Bruijn graph



Collapse the de Bruijn graph



Traverse the graph



Assemble Transcript Isoforms

 ACCGCCCACAGCGCTTCCTGCTGGTCTCTTGTTGGTCGTAG	
 ACCGCCCACAGCGCTTCCT CTTGTTGGTCGTAG	
 ACCGCCCTCAGCGCTTCCTCTTGTTGGTCGTAG	
 ACCGCCCTCAGCGCTTCCTGCTGGTCTCTTGTTGGTCGTAG	

Contrasting Genome and Transcriptome *De novo* Assembly

Genome Assembly

- Uniform coverage
- Single contig per locus
- Assemble small numbers of large Mb-length chromosomes
- Double-stranded data

Transcriptome Assembly

- Exponentially distributed coverage levels
- Multiple contigs per locus (alt splicing)
- Assemble many thousands of Kb-length transcripts
- Strand-specific data available



Trinity Aggregates Isolated Transcript Graphs

Genome Assembly

Single Massive Graph



Entire chromosomes represented.

Trinity Transcriptome Assembly

Many Thousands of Small Graphs



Ideally, one graph per expressed gene.

Trinity – How it works:



Thousands of disjoint graphs



- Decompose all reads into overlapping Kmers => hashtable(kmer, count)
- Read: AATGTGAAAACTGGATTACATGCTGGTATGTC...

AATGTGA	
ATGTGAA	Overlapping kmers of length (k)
TGTGAAA	

Kmer Catalog (hashtable)

Kmer	Count among all reads
AATGTGA	4
ATGTGAA	2
TGTGAAA	1
GATTACA	9



- Decompose all reads into overlapping Kmers => hashtable(kmer, count)
- Identify seed kmer as most abundant Kmer, ignoring low-complexity kmers.

GATTACA 9

Kmer Catalog (hashtable)

Kmer	Count among all reads
AATGTGA	4
ATGTGAA	2
TGTGAAA	1
GATTACA	9



- Decompose all reads into overlapping Kmers => hashtable(kmer, count)
- Identify seed kmer as most abundant Kmer, ignoring low-complexity kmers.
- Extend kmer at 3' end, guided by coverage.





GATTACA 9 T C



GATTACA 9 T C

GATTACA 9 T₀ C

GATTACA 9 C₄ C₄

 $\mathbf{GATTACA}_{9} \qquad \mathbf{C}_{4} \qquad \mathbf{A}_{1} \\ \mathbf{T}_{0} \\ \mathbf{C}_{4} \qquad \mathbf{C}_{4}$

Report contig:AAGATTACAGA....

Remove assembled kmers from catalog, then repeat the entire process.

Expressed isoforms

Chrysalis Re-groups Related Inchworm Contigs

Chrysalis uses (k-1) overlaps and read support to link related Inchworm contigs

Butterfly Example 1: Reconstruction of Alternatively Spliced Transcripts

Reconstruction of Alternatively Spliced Transcripts

Reconstructed Transcripts

Reconstruction of Alternatively Spliced Transcripts

Reconstructed Transcripts

Reconstruction of Alternatively Spliced Transcripts

Butterfly Example 2: Teasing Apart Transcripts of Paralogous Genes

Teasing Apart Transcripts of Paralogous Genes

Strand-specific RNA-Seq is Preferred

Computationally: fewer confounding graph structures in de novo assembly: ex. Forward != reverse complement

(GGAA != TTCC)

Biologically: separate sense vs. antisense transcription

NATURE METHODS | VOL.7 NO.9 | SEPTEMBER 2010 |

Comprehensive comparative analysis of strand-specific RNA sequencing methods

Joshua Z Levin^{1,6}, Moran Yassour^{1-3,6}, Xian Adiconis¹, Chad Nusbaum¹, Dawn Anne Thompson¹, Nir Friedman^{3,4}, Andreas Gnirke¹ & Aviv Regev^{1,2,5}

Strand-specific, massively parallel cDNA sequencing (RNA-seq) is a powerful tool for transcript discovery, genome annotation

Nevertheless, direct information on the originating strand can substantially enhance the value of an RNA-seq experiment. For

'dUTP second strand marking' identified as the leading protocol

computational pipeline to compare library quality metrics from any RNA-seq method. Using the well-annotated *Saccharomyces cerevisiae* transcriptome as a benchmark, we compared seven library-construction protocols, including both published and transcribed strand or other noncoung to tris, demarcate the exact boundaries of adjacent genes transcribed on opposite strands and resolve the correct expression levels of coding or noncoding overlapping transcripts. These tasks are particularly challenging in small microbial genomes, prokaryotic and eukaryotic, in which

dUTP 2nd Strand Method: Our Favorite

Modified from Parkhomchuk et al. (2009) Nucleic Acids Res. 37:e123

Slide courtesy of Joshua Levin, Broad Institute.

Overlapping UTRs from Opposite Strands

Schizosacharomyces pombe (fission yeast)

Antisense-dominated Transcription

Trinity output: A multi-fasta file

Double

0

44.4%

8.5

Name

Read depth

Text outline

>comp0 c0 seq1 len=5528 path=[1:0-3646 10775:3647-3775 3648:3776-5527]

AATTGAATCCCTTTTTGTATCGAAAAATTGAAAGTGAAAGACATATACAGATTGAATGCGGTGATGGAATGCAAATTACGAACATTAGAAAATTACGAAAATTGACGAACATGACGACACCTAGGTTGG TOCACTOCCATCATOTOGAGATACTACAGAGGACTATCCGTCCACAGGACGTAACTGAACCCGATTCCTCCTTTCTTGCAAAGTCTTGACTTGACTAGGATCTCAGTAGAAAAAGCAGCAGCATTCTTTTTTCAGTCT TCACAGTAACTGGACACCCAAAGGACAGAAATAGTCTCAACGAAGAAGACCAGAATTCTCTAGGACTGCGGGTCTTCACATTGCCATCTGTAAGTCTCTAAGAGGTCCCCTTTACATGTCCCGAAGAACACCTCT TOTO AGG TO TOTO A TACALEAGE CONSTRAINT CANCELEGE DE CONSTRUCTION OF TAXATO ANTICONTROL ANTICONTROL AND CONSTRUCTION OF TAXATO ANTICONTROL ANTICONT GCTTCTCCCATACATCAATGAGCACATGAACAGCGAGCAGCAGCAGTAATAGTCTGAGAACTGCAATCCGGTCTCTAAACAACAAGAGCGCCCCAAACCCGTGCTGGTACCTTGAGCAGCACATCCAGTCCGTGTCTTTGACCACATCCAG TCCTGCTGCCAGTTCTCTGTAAAACCAATGGCCTTGAGAACCTTTGCACAGAGATCTTTGTGTTTCTCAACAGTTTATCAGTTGCCATTATCATTCCATTATCAATGGCCCG

>comp0 c0 seq2 len=5399 path=[1:0-3646 3648:3647-5398]

ARTTGRATCCCTTTTTGTATCGRARASCTGRARGCATATACAGATGGATGGATGGATGGGATGGAAATATAATGCARATTAGAAAATTATGAAAATTGATGGAGGACGACGACGACGCCCCGGGTGTGG

TATTCAAAAAGGCCCTTTTTTTGGGGATGGAGCACGTGATACTCTGATGCAACCATGATGTAGGCTCCACACTCCTACAAGTAAGAAAGTACTTTGCACACACT De Bruijn graph information TTTTTGTGAATCOCAGACAGTTACGATAAAGAATGCAATGGTGTGCTGCTGCAGCAGTCCATGGGAAAGACCAGTCCTCACCAAGTCATCTTTCACCTTACAGTTAC Nodes: 279 Edges: 332 GTGRACARCATGARTACACCCTGATGCAGTCTTRAGTGTCRACAGGACACCAACATCAGGCCCATTATARAAACATACCTTTTCAACCTTARAAACCTA Total length: 4,685,914 GCTGAGCTCTCCCAGGAGCGGGGCTGGGGGATGTCTTGGCTGCTGGGGGCTGTGGAGCCAAAACTGAGCACCTGTGGTGGCGCGGGGGCGCGGGGCACGTAC Graph drawing CCCGGTTCTGTGGGATGGCACAAGGAACCTGCCACTGAGGGAAGGGTCCTGCTGCTGCTACACCTGCTTTTGCCTCATACGGCTACAGGATGGGCTACGGCGGCTACGGCGCTACGGCTACGGCTACGGCTACGGCTACGGCTACGGCTACGGCTACGGCTACGGCTA Scope: Entire graph GTARACCCAGATGAGGGTCCTCCTGGTTTATATACARTARTTATTCGTTGTATATATTAATACCCCCARARATTAATTCACGATCARGCCCTCCATGCA Style: Single CAGATOTTOTTTGCTGCGGGGAACGTGCTGCGCTGCGCACCTTCGGTGTTTATGGTTCAGAATCCCTTAAATTTTAGTTCTTTGTACTCCCCA Draw graph TCACAGOTAACTGGACACCCAAAGGATGACAGAAATAGTCTCAACGAAGAAGACCAGGATTCTCTAGGACTGCGGGCTGTGGGATCTCCACATTGCCATCTGTAACTTCT CRATAGOTOTANGATGCAAGACTATTTCTATTCGTTCCTGTATATCCACTGTACCCCCTCTAAGCCACTGTCCCAAGCTCATACCATGTCGCTGACCAGATGC . Node width: AGAAACATCCCTGTCTTTGGATCCCAACTGAAGAAAACATCCAAGTCTGCGGTGTCTGGCAAATCTTCCCATGGAACATCATCTTCCCCTGGCCTCAGATTCAT Random colours TTGCTTCAAGTAGAAGGTCTAACAGCATCCGCTCAGTGCGCTACTTCTGCGAGAAATGGAGAGAATTATTCAGCCTGCTACAAAGCGGATAAACCCTCGGGATCTTCTTC Node labels 0 Lenath ARCARCCCAAGGAGCTGGATGAATGAATGTGTGCACTGGGCGGCCAGGCAAGAGGTCAACGAACACCTTCAGGTCTGTGAAGCAGCATGGCATGCCCCCCAACTT BLAST hits CCASCTTGTATTCGTCATTACAGCCTTGGCTCCGCAAGCGCCCGATGAGCTCCAGCTTAGCTAGGTGTGGGCCTCGGACACGTCGAGAGCTTTGTGATGCCTCTG 0 Font BLAST 0 Create/view BLAST search TGAAAGCTTCTTCCTGCCTGCCAGTTCTCTGTAAACCAATGGCCTTGAGAACCTTTGCACAGTGAAGATCTTTGTGTTTCTCAACAGTTTATCAGCTTGCTGAA TAAATGGGCCGGAGGGCGGCGGTGGTTAGGGTCCTGCACATGGCCCGGGGCGAGAGGCGCAGAACCTCAGT

Can visualize using Bandage

https://rrwick.github.io/Bandage/

• Trinity assembly practical