The University of hong kong

Department of computer science

Faculty of Engineering

RNA-Seq Assembly Using Succinct de Bruijn Graph

Author:
Dai Songcheng

Supervisor:
Dr. Luo Ruibang

September 30, 2018

Contents

1 Introduction

In the field of transcriptome study, full length transcripts assembly plays a significant role. Current trnascriptome assembly strategies can be generally categorized into three types, depending on whether a reference genome is available: a reference based strategy, a *de novo* strategy or a combined strategy [7]. The *de novo* assembly strategy does not require a reference genome, so most organisms without a high-quality genome can use it to produce an initial set of transcripts [7].

With the development of next-generation sequencing(NGS), or RNA sequencing (RNA-Seq), the *de novo* assembly strategies are renovate to be more sensitive and accurate. RNA-Seq is an experimental protocol that uses next-generation sequencing technologies to sequence the RNA molecules within a biological sample in an effort to determine the primary sequence and relative abundance of each RNA. It has a much higher throughput and lower cost compared with the transitional Sanger technology . However, the length of the reads obtained from the common NGS platforms are much shorter than those from Sanger[7]. This leads to a change of the graph representation during assembly. Previously, an overlap-layout-consensus [2] approach is used for the long reads, with one read per node. But this approach is not suitable for the huge amount of short reads produced by the NGS platforms. With one node per read, it will be extremely large and lengthy to construct the overlap graph [10]. Instead, a new approach using de Bruijn graph is adopted. De Bruijn graph does not use node to represent a read, but to represent a k-mer,a word of k nucleotides. Reads are mapped as paths through the graph, going from one word(node) to the next in a determined order. This method tremendously saves the computational resources and time[10].

In recent years, a lot of $deBruijngraph$-based *de novo* transcriptome assembly tools were developed, such as Trinity[4], SOAPdenovo-Trans[9], trans-ABySS [8] and etc, each with their own advantages. This project aims to adapt a metagenomic assembler, MEGAHIT [5], to transctiptome assembly, such that it is competitive with the others.

2 Motivation

MEGAHIT is a NGS *de novo* assembler for assembling large and complex metagenomics data in a time- and cost-efficient manner [5] using succinct de Bruijn graph(a compress data structure of de Bruijn graph). However, it cannot be directly applied to transcriptome assembly for mainly two reasons. First, a transcriptome assembler must be capable to reconstruct the alternative splicing variants from the same gene loci[7]. Current MEGAHIT assembler cannot distinguish the isoforms and can only produce one contig per gene loci. Secondly, transcriptome assembly can be strand specific to resolve the overlapping genes and antisense transcripts. Currently, MEGAHIT does not support that feature either. However, given the outstanding performance of MEGAHIT on the metagenomic data, it has a great potential to be adapt to produce high-quality full-length transcriptome in a time- and memory-efficient manner.

3 Methodology

3.1 Initial assessment

An initial assessment was conducted to evaluate the performance of MEGAHIT on transcriptome assembly. Four species from different classes were selected as samples, namely *M. musculus*(Mammalia), *S. cerevisiae*(Fungi), *Z. mays*(Plantae) and *C. elegans*(Lower eukaryotic).The raw reads data was prepossessed to remove the adapters and contamination before it was fed to the assembly tools, including Trinity, SOAPdenovo-Trans, TransABySS, and MEGAHIT. And the outputs were assessed by rnaQUAST[3]. rnaQUAST is a tool for evaluating RNA-Seq assembly quality and benchmarking transcriptome assemblers using reference genome and gene database. It can assess the outputs from several different assemblers simultaneously.

According to the rnaQUAST report, MEGAHIT had a very good contig precision and recall, but there were two main problems. First, the number of transcripts produced by MEGAHIT was far more less than the other tools, especially for those short ones. The number of long contigs (¿500bp, ¿1000bp) retained from MEGAHIT had litter difference with the others and the average length of the contigs from MEGAHIT were much longer, which indicated MEGAHIT missed a lot of short contigs. Secondly, MEGAHIT performed worse in transcript variants detection. The number of isoforms increased very slightly to the number of genes compared with other tools.

3.2 Mercy k-mer

It is a common practice for *deBruijngraph* based assemblers to filter out all edges that appear less than d (1 in most cases) times in the read set to discard a large number of erroneous edges and save memory[6]. However, it could be risky for metagenome assembly due to the existence of low-abundance species, whose genes have very low coverage. MEGAHIT thus adapts a mercy k-mer strategy to recover those low-depth edges. Given two solid k-mers (i.e. those who are incoming or outgoing vertices of solid edges) x and y from the same read, where x has no out-degree and y has no in-degree in the graph composed by solid k-mers and edges, if no (k+1)-mers between x and y in that read are solid edges, they will be added to the de Bruijn graph as mercy k-mers (mercy edges) [6].

It is believed that the mercy k-mer strategy will also be helpful in transcriptome assembly, because transcripts vary a lot in magnitude and some of them have very low sequencing depth. Mercy k-mer can efficiently protect those unique transcript from being abandoned. A experiment will be conducted by adjusting d and turn on/off mercy k-mer to verify this hypothesis.

3.3 Multiple k-mer sizes

The size of k greatly determines the construction of the de Bruijn graph, thus prominently influences the output of the assembler. Small k-mers can increase the connectivity and the overlapping of the graph, making it easier to fill up the gaps in low coverage regions. While larger k is helpful to resolve longer repeats[6]. To address the trade-off, MEGAHIT adopts a multiple k-mer sizes strategy. It starts from $k = k_{min}$, constructs the de Bruijn graph of edge size $k+1$, conducts graph cleaning (tip removal, bubble merging and local low depth edges removal), produces contigs for this iteration, adds "step" to current k, selects edges ($k+s+1$ mers) for next graph, and then constructs next graph for edge size $k+s+1$ for next iternation. This procedure will repeat until $k = k_{max}$. However, the default value of the step s is set to be 20, which may be a bit large for transcriptome assembly. The length of RNA-Seq reads is generally between $50-150bp$ [1], this step value will cause very low connectivity after 2-3 iterations. Therefore, experiments will be conducts to tweak the step value and k_{min}/k_{max}.

3.4 Graph Cleaning Strategy and strand specific feature

In each iteration, graph cleaning is conducted. But in MEGAHIT, the long and similar bubbles are merged and only one path is reserved[6].This impedes MEGAHIT from identifying the transcript isoforms. Further research and experiments on graph manipulation will be conducted to enable MEGAHIT to figure out alternative splicing variants.

Besides, strand specific feature is supported by many transcriptome assembly tools. A new model will be added to enable strand specific option.

4 Project Schedule

Index	Task	Begin	End
1	Initial assessment	done	done
2	Mercy k-mer	2018/9/15	2018/10/1
3	Multi k-mer size	2018/10/1	2018/10/14
4	Graph Cleaning	2018/10/15	2018/11/30
5	Strand Specific	2018/12/1	2019/1/1
6	Further exploration	2019/1/1	2019/4/15

References

[1] "Considerations for rna-seq read length and coverage," Jun 2018. [Online]. Available: https://support.illumina.com/bulletins/2017/04/considerations-for-rna-seq-read-length-and-coverage-.html

[2] S. Batzoglou, "Algorithmic challenges in mammalian genome sequence assembly," *Encyclopedia of genomics, proteomics and bioinformatics. John Wiley and Sons*, 2005.

[3] E. Bushmanova, D. Antipov, A. Lapidus, V. Suvorov, and A. D. Prjibelski, "rnaquast: a quality assessment tool for de novo transcriptome assemblies," *Bioinformatics*, vol. 32, no. 14, pp. 2210–2212, 2016.

[4] M. G. Grabherr, B. J. Haas, M. Yassour, J. Z. Levin, D. A. Thompson, I. Amit, X. Adiconis, L. Fan, R. Raychowdhury, Q. Zeng, *et al.*, "Full-length transcriptome assembly from rna-seq data without a reference genome," *Nature biotechnology*, vol. 29, no. 7, p. 644, 2011.

[5] D. Li, C.-M. Liu, R. Luo, K. Sadakane, and T.-W. Lam, "Megahit: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de bruijn graph," *Bioinformatics*, vol. 31, no. 10, pp. 1674–1676, 2015.

[6] D. Li, R. Luo, C.-M. Liu, C.-M. Leung, H.-F. Ting, K. Sadakane, H. Yamashita, and T.-W. Lam, "Megahit v1. 0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices," *Methods*, vol. 102, pp. 3–11, 2016.

[7] J. A. Martin and Z. Wang, "Next-generation transcriptome assembly," *Nature Reviews Genetics*, vol. 12, no. 10, p. 671, 2011.

[8] G. Robertson, J. Schein, R. Chiu, R. Corbett, M. Field, S. D. Jackman, K. Mungall, S. Lee, H. M. Okada, J. Q. Qian, *et al.*, "De novo assembly and analysis of rna-seq data," *Nature methods*, vol. 7, no. 11, p. 909, 2010.

[9] Y. Xie, G. Wu, J. Tang, R. Luo, J. Patterson, S. Liu, W. Huang, G. He, S. Gu, S. Li, *et al.*, "Soapdenovo-trans: de novo transcriptome assembly with short rna-seq reads," *Bioinformatics*, vol. 30, no. 12, pp. 1660–1666, 2014.

[10] D. Zerbino and E. Birney, "Velvet: algorithms for de novo short read assembly using de bruijn graphs," *Genome research*, pp. gr–074 492, 2008.