Metagenomics Assemblers Evaluation [Or Whatever Titus suggests :)]

author[1], Sherine Awad [In whatever order and with whoever should be added] [2], author[3,*]
2 Author1 Dept/Program/Center, Institution Name, City, State, Country
3 Same as Titus Departments/Program/Center, Institution Name, City, State, Country
4 Author3 Dept/Program/Center, Institution Name, City, State, Country
∗ **E-mail: Corresponding author@institute.edu**

Abstract

Author Summary

Introduction

Materials and Methods

Datasets

Podar (write correct name) datasets where downloaded from XX. The dataset represent XX brief description for the data.

Pre-assembly Treatments

We assembled the reads using a combination of different preprocessing and assembly approaches. The preprocessing treatments are:

1. **Quality Filtering:** In this treatment, low quality bases were trimmed and low quality reads were removed using trimmomatic [**?**]. After quality trimming reads were either directly assembled, or first preprocessed with digital normalization and then assembled. The original datasets contains 5536289548 base pairs and 54814748 sequences in the left pair and 5536289548 base pairs and 54814748 sequences in the right pair.

   After quality filtering, the paired-ended file contains 10547795822 base pairs 104433622 sequences while the single-ended file contains 184437913 base pairs and 1893243 sequences.

2. **Digital Normalization:** Digital normalization works after sequencing data has been generated, progressively removing high-coverage reads from shotgun data sets. This normalizes average coverage to a specified value, reducing sampling variation while removing reads, and also removing the many errors contained within those reads. This data and error reduction results in dramatically decreased computational requirements for de novo assembly. Moreover, unlike experimental normalization where abundance information is removed prior to sequencing, in digital normalization this information can be recovered from the unnormalized reads [**?**] After digital normalization, the pair ended file contains 1687588894 base pairs and 16853716 sequences while the single ended file contains 5859253 base pairs and 64638 sequences.

3. **Partitioning:** In this treatment, we partitioned the filtered data set based on de Bruijn graph connectivity and assembled each partition independently. Subsequently, partitioning separates reads based on transitive connectivity, resulting in easily assembled subsets of reads.

4. **Reinflation:**

Metagenomes Assembly

We assembled the reads using four different assemblers; Velvet [**?**], Idba [**?**], Spades [**?**], and Megahit [**?**] in combination with different preprocessing treatments.

Results

Metagenomes Metrics

Table 1 shows various quality metrics for the results of the assembly using combinations of four different assemblers and different preprocessing treatments. Table 2 shows the percentage of unaligned sequences when mapping the raw reads to the results assembly.

Table 1. Assembly Quality Metrics

Treatment/Quality Metric	Quality Filtering	Digital Normalization	Partition	Reinflation
(1) Velvet				
Genome Fraction	72.949	89.043	88.879	-
Unaligned Length	8,977,149	10,909,693	11,317,834	-
Misassembled contigs length				
N50	38028	18944	8504	-
(2) Idba				
Genome Fraction	90.969	91.003	90.082	88.346
Unaligned Length	10,709,716	10,637,811	10,644,357	10,288,486
Misassembled contigs length				
N50	4,977,3	4,782,8	2,657,5	2,984,0
(3) Spades				
Genome Fraction	90.424	90.173	89.272	89.798
Unaligned Length	10,597,529	10,621,398	10,500,235	10,461,672
Misassembled contigs length				
N50	4,277,3	3,558,0	2,231,9	2,698,9
(4) Megahit				
Genome Fraction	89.961		88.769	
Unaligned Length	10,525,444		10,565,036	
Misassembled contigs length				
N50	3,176,9		1,539,3	

Resources Requirements Reduction based on Digital Normalization, Partitioning

Table 3 shows the running time and memory utilizations for four assemblers and different reads treatments. The results assembly after digitial normalization, partitioning, and reinflation show a significant decrease in running time and memory utilization. For Idba, digital normalization reduces 27 hours in the running

Table 2. Reads Mapping

Treatment/Quality Metric	Quality Filtering	Digital Normalization	Partition	Reinflation
(1) Velvet				
No. of Unaligned Sequences				-
(2) Idba				
No. of Unaligned Sequences	2092523	2761871	3602590	
(3) Spades				
No. of Unaligned Sequences	3013782		3579651	
(4) Megahit				
No. of Unaligned Sequences				

time. While partitioning reduces 25 hours in the running time. For SPAdes, digital normalization reduces 52 hours in the running time. While partitioning reduces 59 hours in the running time. For Velvet, digital normalization reduces 54 hours in the running time. While partitioning reduces 56 hours in the running time. For Megahit, digital normalization reduces XX hours in the running time. While partitioning reduces XX hours in the running time.

Digital normalization and Partitioning also reduce memory requirements. For Idba, digital normalization reduces XX KB of memory utilization. While partitioning reduces XX KB of memory utilization. For SPAdes, digital normalization reduces XX KB of memory utilization. While partitioning reduces XX KB of memory utilization. For megahit, digital normalization reduces XX KB of memory utilization. While partitioning reduces XX KB of memory utilization.

Reinflation Increases Memory Requirements

Table 3. Running Time and Memory Utilization

Treatment/Quality Metric	Quality Filtering	Digital Normalization	Partition	Reinflation
(1) Velvet				
Running Time	60:42:52	6:48:46	4:30:36	-
Memory Utilization in KB	1594851536	827412304	1156729920	-
(2) Idba				
Running Time	33:53:46	6:34:24	6:34:24	2:33:17
Memory Utilization in KB	129853424	104736448	93584624	393938608
(3) Spades				
Running Time	67:02:16	15:53:10	7:54:26	6:50:46
Memory Utilization in KB	400340512	127423856	129715072	434531888
(4) Megahit				
Running Time			0:00:27	
Memory Utilization in KB			7548668	

Reads Mapping

We estimated the percentage of unaligned sequences by each assembly treatment and using the four assemblers. We mapped the raw reads to each assembly. Then we extracted the unaligned sequences to

each assembly. Table XX shows the percentages of unaligned sequences from raw reads to each assembly treatment using the four assemblers under study. For all treatments assemblies, the full set of trimmed reads were used for mapping. Default parameters were used, and paired reads were mapped only.

Samtools [**?**] was used for format conversion from SAM to BAM format, and also to calculate the percentage of mapped reads.

Discussion

Main points Explain the significance of each treatment

Acknowledgments

References

@articleBrown2012, Author = C Titus Brown and Adina Howe and Qingpeng Zhang and Alexis B Pyrkosz and Timothy H Brom, Journal = , Month = May, Number = , Title = A reference-free algorithm for computational normalization of shotgun sequencing data, Volume = , Year = 2012

Figure Legends

Tables

Supporting Information Legends