

Viral Reads Assembly Enhancer (ViRAE)

User manual

28 February 2024, version 1

TABLE OF CONTENTS

VIRAE methodology description.....	3
VIRAE standalone application (offline).....	4
Installation	4
Execution.....	4
Run examples.....	5
Output.....	6
VIRAE web-based application (online).....	8

VIRAE methodology description

Viral Reads Assembly Enhancer (VIRAE) is a context-based trimming bioinformatics tool, especially designed for viral metagenomics, which allows Next Generation Sequencing (NGS) read decontamination based on any given reference sequence(s). VIRAE is powered by an updated version of [Zero-Waste Algorithm \(ZWA\)](#) and incorporates ready-to-use well-established bioinformatics software to detect and dissect partially mapped reads (chimeric reads) by specifically removing the moieties, which align to the given reference sequence(s). The clean output reads enhance *de novo* assembly performance, increasing the availability of reads for more accurate and more efficacious *de novo* virus genome assembly. The concept behind the VIRAE pipeline is outlined in **Figure 1**.

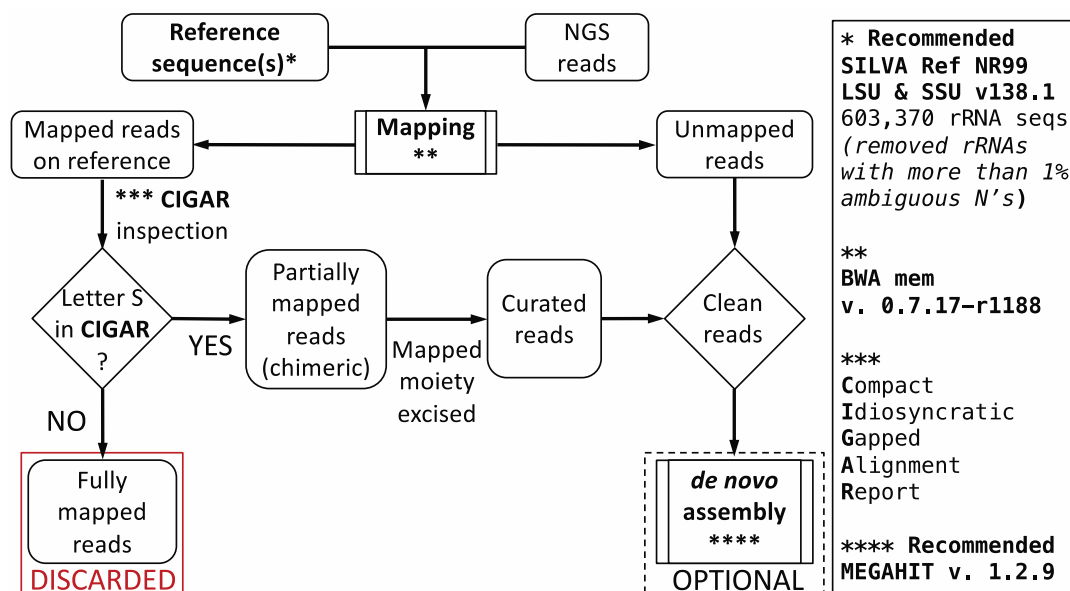


Figure 1

Important note: VIRAE focuses on the identification and decontamination of partially mapped (chimeric) NGS reads on any given reference sequence(s). For optimum decontamination, we highly recommend downloading and inputting [our custom SILVA ribosomal database \(RiDB\)](#) as reference file, containing 603370 seqs of 16S, 18S, 23S and 28S rRNAs from a wide variety of *Archaea*, *Bacteria* and *Eukarya* organisms. Noteworthy, *de novo* assembly of the clean output reads after decontamination is optional, and therefore is not included in the main VIRAE pipeline or the program's prerequisites. However, the user may separately perform *de novo* assembly on the clean output reads using the software of preference, with [MEGAHIT *de novo* assembler](#) being highly recommended.

VIRAE standalone application (offline)

Installation

The VIRAE standalone application is a Bash shell script distributed for Linux and MacOS systems and may be executed directly after making the downloaded VIRAE.sh file executable (e.g. command 'chmod +x'). The prerequisites of VIRAE ([bwa 0.7.17](#) and [samtools 1.13](#)) will be verified for installation upon VIRAE execution and if not installed, they will be downloaded automatically by the program.

Execution

The parameters of the VIRAE standalone application are summarized in the following table:

Required arguments	Description	Deployment
-i <string>	directory of INPUT NGS READS file (.fastq, .fq or .gz extension)	Full
-r <string>	directory of INPUT REFERENCE file (.fasta, .fa, .fna, .fsta or .gz extension)	Full
-m <string>	directory of MAPPED NGS READS on REFERENCE file (.bam extension)	Partial
-o <string>	directory of OUTPUT folder	Full & Partial
Optional arguments	Description	Deployment
-l <integer>	alignment stringency value (default value 30 <30 loose, >30 stringent)	Full
-u <string>	directory of UNMAPPED NGS READS on INPUT REFERENCE file (.bam, .fastq, .fq or .gz extension)	Partial
-t	run VIRAE on a test dataset to verify installation	TEST MODE

Run examples

In order to better comprehend the use and output of the VIRAE standalone application, we highly recommend inputting the -t flag only the first time you run it so as to deploy VIRAE on a small test dataset, which also verifies the installation of all prerequisites and downloads them automatically if needed.

VIRAE test mode run example:

```
./VIRAE.sh -t
```

For non-test run, the VIRAE standalone application may be fully or partially deployed upon execution depending on the available user input files. In the case of VIRAE full deployment, the directories of NGS reads (FASTQ format) and appropriate reference file (FASTA format) must be provided as arguments after the -i and -r flags respectively, in order to be able to perform all necessary alignments. The user also has the ability to adjust the mapping sensitivity of the incorporated BWA software by passing the desired level of alignment stringency as an integer number after the -l flag (default value 30 | <30 loose, >30 stringent).

Full VIRAE deployment run examples:

```
./VIRAE.sh -i reads.fastq -r ref.fasta -o ./
```

```
./VIRAE.sh -i reads.fq.gz -r ref.fasta.gz -o ./
```

```
./VIRAE.sh -i reads.fq.gz -r ref.fasta.gz -l 40 -o ./
```

For the alternative and faster partial deployment of VIRAE, in which the user may have already carried out the desired alignment with the mapping software of preference, the directory of a BAM file may only be provided after the -m flag, instead of FASTQ and FASTA files. Alongside the input BAM file, the user may optionally pass the output unmapped reads of the performed alignment in FASTQ or BAM format after the -u flag, for later use by the algorithm.

Partial VIRAE deployment run examples:

```
./VIRAE.sh -m mapped.bam -o ./
```

```
./VIRAE.sh -m mapped.bam -u unmapped.bam -o ./
```

```
./VIRAE.sh -m mapped.bam -u unmapped.fq.gz -o ./
```

Output

VIRAE outputs 2 files, which are: i) the clean reads after processing as a GZIPPED FASTQ file with the suffix "*VIRAE_cleaned.fastq.gz*", and ii) a detailed cleaning report file named "*VIRAE_cleaning_report.out.gz*". The generated GZIPPED FASTQ file contains all the clean reads by VIRAE and may be used separately for *de novo* assembly or other downstream analysis. The generated report file is a multi-column file, which provides further information and details on the cleaning performed by VIRAE, in the following format:

Column header	Description
Read_ID	Unique read sequence identifier
RefID	Unique reference sequence identifier
CIGAR	Compact Idiosyncratic Gapped Alignment Report string
Read_seq	Complete read sequence
Read_seqlength	Total read sequence length
Mapped_seq	Mapped sequence of read
Mapped_seq_length	Mapped sequence length of read
Mapped_start	Mapping start position of read
Mapped_end	Mapping end position of read
Left_unmapped_seq_start	New start position of read sequence after left-side trimming
Left_unmapped_seq_end	New end position of read sequence after left-side trimming
Left_unmapped_seq	New read sequence after left-side trimming
Left_unmapped_seq_quality	New read sequence quality after left-side trimming
Left_unmapped_seqlength	New read sequence length after left-side trimming

Right_unmapped_seq_start	New start position of read sequence after right-side trimming
Right_unmapped_seq_end	New end position of read sequence after right-side trimming
Right_unmapped_seq	New read sequence after right-side trimming
Right_unmapped_seq_quality	New read sequence quality after right-side trimming
Right_unmapped_seqlength	New read sequence length after right-side trimming

An overall summary of the VIRAE analysis is also provided at the end of the generated report file, which has the following line-by-line format:

Line header	Description
BWA alignment stringency	Alignment stringency value of BWA (default value 30 <30 loose, >30 stringent)
Total input reads	Total number of user input reads
Total unmapped reads	Total number of unmapped reads
Total mapped reads	Total number of mapped reads
Fully mapped reads	Total number of fully mapped reads only
Fully mapped reads / Total mapped reads	Percentage of the total number of fully mapped reads to total number of mapped reads
Partially mapped (chimeric) reads	Total number of chimeric reads only
Partially mapped (chimeric) reads / Total mapped reads (%)	Percentage of the total number of chimeric reads to total number of mapped reads
Average mapped bases	Numnber of average mapped bases in chimeric reads
Average mapped bases / Average read length (%)	Percentage of the number of average mapped bases in chimeric reads to average length of chimeric reads
VIRAE cleaned chimeric reads	Number of chimeric reads cleaned by VIRAE
VIRAE cleaned chimeric reads / Chimeric reads (%)	Percentage of the number of chimeric reads cleaned by VIRAE
Total clean reads (Unmapped + VIRAE cleaned)	Total number of clean reads (equal to the sum of unmapped reads + chimeras cleaned by VIRAE)
VIRAE discarded chimeric reads	Number of chimeric reads discarded by VIRAE

VIRAE discarded chimeric reads / Chimeric reads (%)	Percentage of the number of chimeric reads discarded by VIRAE to total number of chimeric reads
Total discarded reads (Fully mapped+VIRAE discarded)	Total number of discarded reads (equal to the sum of fully mapped reads + chimeras discarded by VIRAE)
Execution time (seconds)	Total execution time of VIRAE (wall clock run time)

VIRAE web-based application (online)

Apart from the VIRAE standalone application, the user may utilize the [VIRAE online tool](#), which does not require the installation of any software but solely the provision of the appropriate input files according to the following 3 steps:

- A) Deployment method selection:** Similarly to the standalone application, the user may choose to fully or partially deploy VIRAE depending on the available input through our online platform. If NGS reads and reference files are available in FASTQ and FASTA formats respectively, then the user should choose “Method 1” as displayed in **Figure 2**, which corresponds to full VIRAE deployment. Alternatively, if the user has already performed the desired alignment between the NGS reads and reference file of preference, then “Method 2” should be selected, which stands for the faster partial deployment of VIRAE, with the sole input of the appropriate BAM file.

ViRAE upload method selection

Method 1	?	Full ViRAE deployment
Method 2	?	Partial ViRAE deployment

Figure 2




- B) Input files upload:** Clicking on the VIRAE deployment method of preference, redirects the user to the upload webpage. Upon selection of full VIRAE deployment (Method 1), the

webpage displays three different upload options to choose from, for the necessary FASTQ and FASTA files separately. These upload options, as displayed in **Figure 3**, are:

1. selection of FASTQ or FASTA input file from a prompt file dialog,
2. submission of a valid SRA accession number (in the case of FASTQ input) or selection from a dropdown menu list of recommended reference files (in the case of FASTA input), or
3. provision of the appropriate link address, where the FASTQ or FASTA input file is stored.

ViRAE upload page

Upload input NGS reads file (supported formats FASTQ or GZ)

1	<input type="radio"/> Select from file dialog  <input type="button" value="Choose file"/> No file chosen	2	<input type="radio"/> SRA accession number from NCBI SRA database  <input type="text"/>	3	<input type="radio"/> Direct download link address from Filetransfer  <input type="text"/>
----------	--	----------	---	----------	--

Upload reference file (supported formats FASTA or GZ)




1	<input type="radio"/> Select from file dialog  <input type="button" value="Choose file"/> No file chosen	2	<input type="radio"/> Select from a list of recommended references  SILVA SSU+LSU rRNA (v138.1) ▼	3	<input type="radio"/> Direct download link address from Filetransfer  <input type="text"/>
----------	--	----------	---	----------	--

Figure 3

As regards to the partial VIRAE deployment (Method 2), there are two available upload options, as displayed in **Figure 4**, which are:

1. selection of the necessary BAM input file from a prompt file dialog, or
2. provision of the appropriate link address, where the necessary BAM input file is stored.

ViRAE upload page

Upload BAM file (supported formats BAM)



1	<input type="radio"/> Select from file dialog  <input type="button" value="Choose file"/> No file chosen	2	<input type="radio"/> Direct download link address from Filetransfer  <input type="text"/>
----------	--	----------	--

Figure 4

Submission of the required input files triggers the upload process and redirects to a new webpage, where the user is informed about the upload progression in real time. In case of upload failure, the user is redirected automatically back to the upload webpage after clicking "OK" on the prompted warning message.

- C) VIRAE implementation and output:** After successful upload of the appropriate input files, the back-end script execution of VIRAE begins and the user is informed about its

progression in real time as displayed in **Figure 5**. Upon VIRAE run completion, an overall summary is displayed at the current webpage, along with a download link corresponding to a zipped folder containing the clean reads and generated report files by VIRAE.

ViRAE execution result

Running ViRAE now, please wait...

```
##### ViRAE #####  
  
Executing ViRAE...  
  
Verifying installation of ViRAE prerequisites (bwa & samtools), please wait...  
Checking for 0.7.17 version of bwa...  
bwa 0.7.17 correctly installed !!!  
Checking for 1.13 version of samtools...  
samtools 1.13 correctly installed !!!  
  
Performing default BWA alignment, please wait...  
  
Alignment results  
  
Total reads:      100  
Mapped reads:    0 [80 (80.00%) fully mapped + 20 (20.00%) partially mapped/chimeric]  
Unmapped reads:  0  
  
Cleaning chimeric, please wait...  
  
Cleaning results  
Discarded reads: 0 (0.00% of chimeric reads)  
Cleaned reads:   20 (100.00% of chimeric reads)  
ViRAE clean reads: 20  
  
ViRAE script completed successfully !!!
```

Real time ViRAE report

[Download ViRAE files](#)

Download link of ViRAE output files

Figure 5