1 Functionality

This pipeline evaluates the quality of a *de novo* assembled transcriptome and returns an improved version of the transcriptome. The pipeline operates as indicated in Figure 1. The pipeline begins and ends with a run of BUSCO. This tool searches for a set of single-copy orthologs that are present in all insect transcriptomes. The tool returns the number of orthologs that were found in the transcriptome to evaluate completeness.

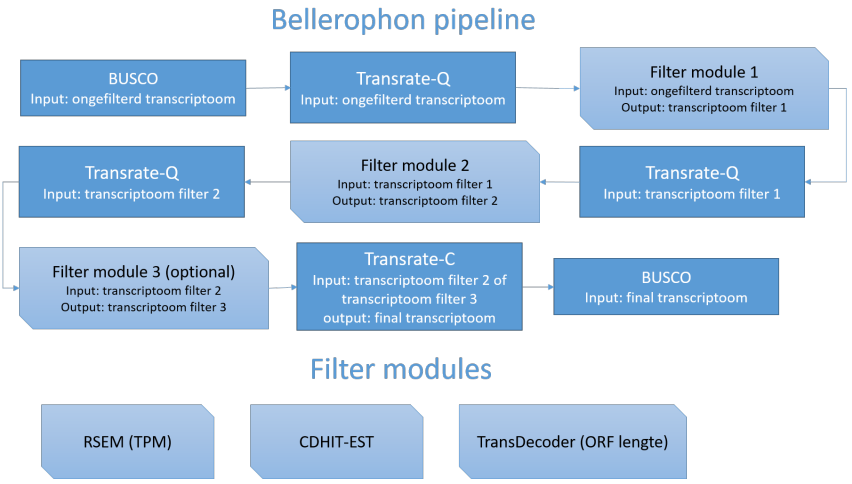


Figure 1: Bellerophon workfow

To evaluate the overall quality of the transcriptome, the tool TransRate is used. To differentiate between TransRate as a measuring tool and TransRate as a filtering step, TransRate as a measuring tool is refered to as TransRate-Q. TransRate as a filtering step is refered to as TransRate-C. TransRate uses read alignment to calculate different quality scores (Figure 2). All quality scores combined form a general quality score for the assembly. This tool is executed before and after each filtering step to evaluate the consequence of the filtering step. Depending on the user's needs, either two or three filter modules are used. These can be arranged in several orders This pipeline filters a *de novo* transcriptome in either two or three ways, chosen by the user. The transcriptome is filtered using CDHIT-EST. This tool clusters highly similar transcripts and chooses one representative transcript. The second filtering is based on expression level. Using the RSEM tool, normalized expression values (TPM) are calculated. The user can execute a third filtering method if necessary. This filtering method is not used by default. This method filters the transcripts based on the length of predicted Open Reading Frames. The tool TransDecoder provides a set of ORFs that meet the cut-off length, regardless of coding potential.
To assess the quality of the transcriptome, the tool TransRate is used. This tool

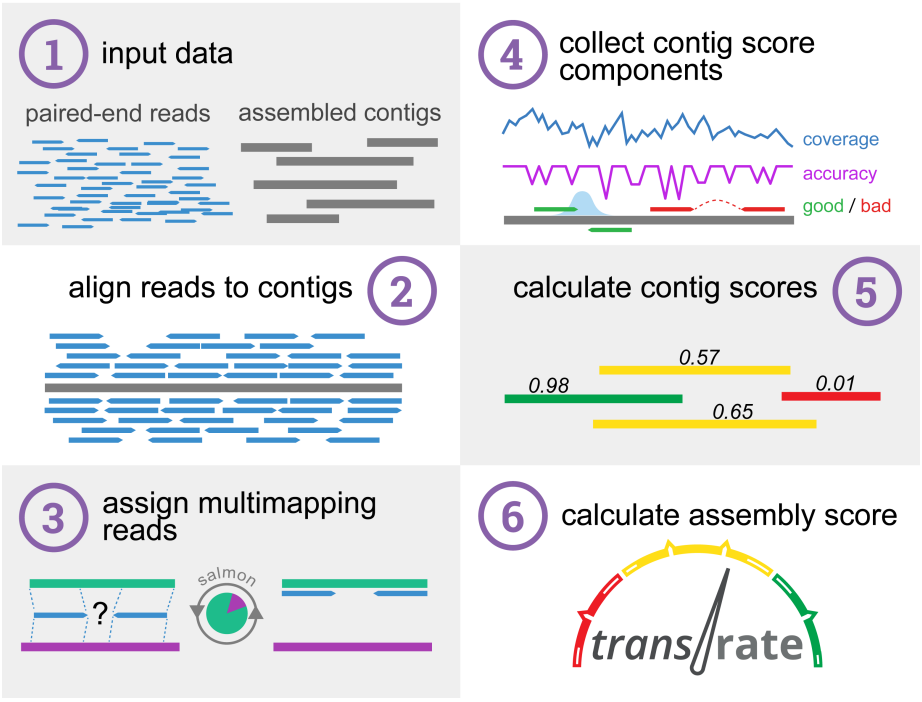


Figure 2: TransRate uses unique read alignment to collect contig quality scores. Every contig is assigned a quality score between 0 and 1. All contig scores combined are used as an overall assembly quality score.

requires the paired end reads and the transcriptome. It calculates a quality score for the assembly and provides several statistics on the reads, the transcriptome and the quality of the transcriptome

Prerequisites

This pipeline uses several tools. These tools can be divided in provided tools and dependent tools. The provided tools are:

1. TransRate
2. RSEM (as part of Trinity)
3. BUSCO
4. TransDecoder
5. faSomeRecords

These tools can be found in the utils directory.
The dependent tools are:

1. CDHIT-EST
2. Python
3. Unix environment
4. Bash

The dependent tools need to be installed on the user's system before the pipeline is able to execute properly. Along with these requirements, the pipeline needs at least 150 GB of free disk space. The alignment files created by TransRate are very large. These files are automatically removed by the pipeline (can be counteracted by a flag) but the space has to be available during the execution of the pipeline. Also, RSEM creates a large BAM file that is not removed by the pipeline. This file can safely be removed once RSEM is done running to free up the disk space. Multithreading is possible for most used tools. The more threads are used, the faster the pipeline will run. Some tools require a good amount of working memory. It is advised to have a minimum of 16 GB of RAM.

Starting the pipeline

Testing the pipeline

To test if all software requirements are met, open a terminal window (in Ubuntu, in the left taskbar click on the uppermost icon and type "terminal" and select the first program). Type

```
cdhit-est -h
```

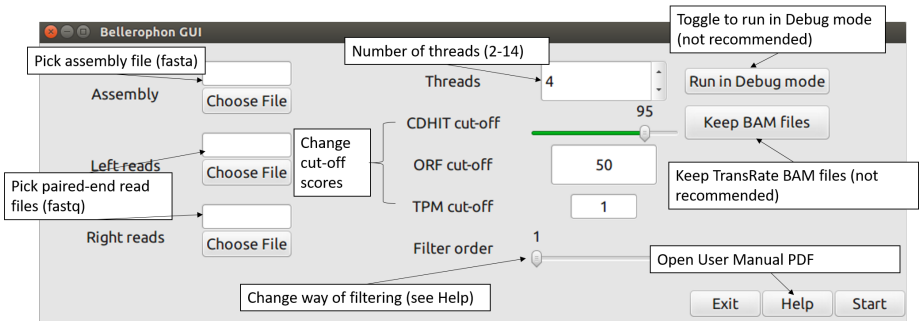
if a message pops up saying "command not found", execute the following command:

```
sudo apt-get install cdhit-est
```

If prompted for a password, provide it.
To test the full pipeline with a demo dataset type:

```
bash Bellerophon_startscript.sh -a Demo.fasta
        -l demo_reads_R1.fastq -r demo_reads_R2.fastq
        -O 4
```

This command runs the pipeline with the provided demo set.
To test the Graphical User Interface (GUI), double-click on the Bellerophon_GUI shortcut. Load the pipeline as indicated in Figure 3 with Demo.fasta as assembly and demo_reads_R1.fastq and demo_reads_R2.fastq. If the pipeline executes without any errors (check the output directory and the log subdirectory), you are good to go.



Running the Commandline version of Bellerophon

To run the commandline version of Bellerophon, the user needs to execute the following base command:

```
bash Bellerophon_startscript.sh -a assembly.fasta
        -l Reads_R1.fastq -r Reads_R2.fastq
```

Where assembly.fasta is the user's assembly and the Reads.fastq files are the paired-end read files.
In this manner, Bellerophon will execute a default run using 4 threads, default cut-off scores and the order: TPM-CDHIT-TransRate. To change the parameters, use the following:

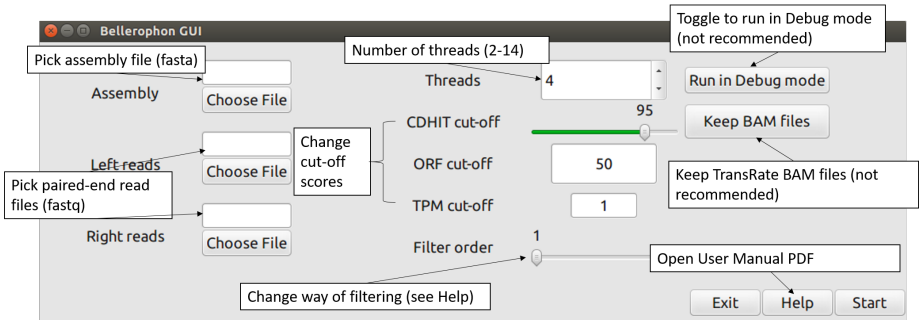
- -c/–cdhit_cutoff CDHIT-EST identity cutoff. 0.95 by default
- -t/–tpm_cutoff Expression cut-off. 1 By default.
- -o/–orf_cutoff ORF length cutoff. 50 aa by default
- -T/–threads number of threads to use. 4 By default
- -D/–debug run in Debug Mode (prints out parameter information). False by default
- -O/–order Filtering order. 1 By default. (Table 1)
- -k/–keep_bam the pipeline doesnt remove the large alignment files made by TransRate. False by default

The filtering order can be managed with the -O flag. The different orders can be used as follows:

Filtering code	Filtering order
1	TPM, CDHIT-EST
2	CDHIT-EST, TPM
3	CDHIT-EST, ORF, TPM
4	CDHIT-EST, TPM, ORF
5	ORF, CDHIT-EST, TPM
6	ORF, TPM, CDHIT-EST
7	TPM, CDHIT-EST, ORF
8	TPM, ORF, CDHIT-EST

Running Bellerophon using the User Interface

To execute Bellerophon with more ease, the user can choose to use the Bellerophon GUI (Graphical User Interface). To open this GUI, double-click the Bellerophon_GUI executable in the main directory. This will open up a screen looking as follows:
To use the software, pick the assembly- and read files using their respective



'Choose File' buttons. This is all that is needed to do a default run, using 4 threads, default cut-off scores and an order of TPM-CDHIT-TransRate. To change these default settings, use the respective buttons, input fields and sliders to adjust to liking. Press the 'Start' button to start the pipeline with the given input values.
A new screen will pop up giving an indication as of what the pipeline is up to. To terminate the pipeline, click the "Cancel Task" button. After the pipeline is done, the message on the screen will display so, and the "Exit" button can now be clicked safely.

Output directory

The output directory of your Bellerophon run will be named after the date and time the pipeline started. This name can safely be renamed **after the pipeline has finished or has been terminated**.

Inside this directory, several files and directories can be found. These directories are (in decreasing order of importance):

- Fasta_files Contains all fasta files that are a product of the pipeline. Most importantly, the final product: Assembly_BEL.fasta (where Assembly is the name of your input assembly
- TransRate_output Contains all different TransRate runs and their respective output files.
- TPM_output Contains all TPM related output. **Note: when using Bellerophon, a very large BAM file will be present in this directory. You can safely delete this after the pipeline is done.**
- Log Contains all log files for the pipeline run. If something goes wrong, the .error files (can be opened in text editors like notepad) indicate what went wrong.
- Run_BUSCO (start and final) Contains the BUSCO output. Most importantly, the summary file containing concrete figures on the assembly's performance.
- RSEM_output RSEM files of (generally) no significant importance to the user
- Bowtie2_files Bowtie2 index files with no significant importance to the user

The main output directory contains the file TransRateResults.csv. This file can be opened using text editors or spreadsheet software and contains the output table for all TransRate runs. This file can easily be used by R to create plots and graphs.