Hyper-Editing Tool Manual

Gili Wolf

Contents

1 Introduction

Hyper-edited RNA detection tool based on the algorithm presented in:

> ***A genome-wide map of hyper-edited RNA reveals numerous new sites***
> Hagit T. Porath, Shai Carmi & Erez Y. Levanon
> Nature Communication 2014 (https://www.nature.com/articles/ncomms5726)

2 Getting Started

2.1 Installation

Hyper-Editing tool is avaiable on: https://github.com/GiliWolf/HE_scripts.git.

2.1.1 Nextflow

The outline of the pipeline for the *Hyper-Editing* tool is constructed using Nextflow. In order to successfully run the tool the installation of Nextflow is requiared.
Please refer to https://www.nextflow.io/docs/latest/install.html

2.2 Main Workflow

Hyper-Editing main workflow consists 3 steps:

1. *Genome Transformation* - Transformation of the genome (12 transformations for each possible base combination) and indexing the transformed genomes (using STAR).
2. *Pre-Analysis* - The main algorithm of the original pipeline. This part involves aligning the samples to the original genome, then transforming the unmapped reads and mapping them again to the transformed genome. Finally, the original sequences of the transformmed reads from the second alignment are recovered and placed in the alignment BAM file.
3. *Detection* - Analyze and detect the editing sites and mismatches events, and filter to find hyper editing reads.

```
basic run:
./nextflow -c HE_scripts/MAIN_HE_SCRIPT.nf.config run HE_scripts/MAIN_HE_SCRIPT.nf --
    run_mode {transformGenome | align | detect | align-detect}
```

NextFlow Basic Options

-c Path to the configuration file.

run Path to the Nextflow script to be executed.

-bg Run in the background (should be used with `nohup`).

-resume Resume the last running process.

2.3 *Hyper-Editing* Run Modes

2.3.1 transformGenome

Executes only *Genome Transformation*. basic option:

```
./nextflow HE_scripts/MAIN_HE_SCRIPT.nf -c HE_scripts/MAIN_HE_SCRIPT.nf.config --
    run_mode transformGenome --genome_fasta <PATH_TO_GENOME> --genome_setup_outdir <
    TRANSFORM_GENOME_OUTDIR_PATH>
```

–genome_fasta Path to the original genome FASTA file, which will be transformed.

–genome_setup_outdir The output directory where the transformed genome will be stored. Each transformed file is prefixed with "$< ref\text{-}base > \_ < alt\text{-}base >$" according to the specific transformation applied. The transformed genome will be placed in the "transformed_genome" directory, and the transformed index will be located in the "genome_index" directory. Both directories can be modified using the `--transform_genome_output_dir` and `--index_output_dir` options, respectively.

2.3.2 align

Executes only *Pre-Analysis*. basic option:

```
./nextflow HE_scripts/MAIN_HE_SCRIPT.nf -c HE_scripts/MAIN_HE_SCRIPT.nf.config --
    run_mode align --align_reads_dir <READS_FASTQ_PATH> --align_outdir <
    ALIGN_OUTPUT_PATH> --genome_index_dir <ORIGINAL_GENOME_INDEX_PATH> --
    transform_genome_dir <TRANSFORMED_GENOMES_DIR> --pair_end {0,1}
```

–align_reads_dirr Directory containing the input FASTQ files for alignment.

–align_outdir Directory where intermediate and output files will be saved.

–genome_index_dir Path to the original genome index directory used for the first alignment.

–transform_genome_dir Directory of the transformed genomes used for the second alignment. (should be the same as –genome_setup_outdir)

–pair_end Indicates whether the reads are single-end (0) or paired-end (1).

2.3.3 Detect

This section executes only *Detection*. It operates in two modes:

1. Continual: Input BAM files are generated using *Pre-Analysis*.
2. Independent: Input BAM files are sourced externally. Please refer to 5.4.2 for more details.

basic option:

```
./nextflow HE_scripts/MAIN_HE_SCRIPT.nf -c HE_scripts/MAIN_HE_SCRIPT.nf.config --
    run_mode detect --detect_input_dir <DETECT_INPUT_PATH|ALIGN_OUTPUT_DIR_PATH> --
    detect_outdir <DETECT_OUTPUT_PATH> --genome_fasta <PATH_TO_GENOME> --
    genome_index_dir <ORIGINAL_GENOME_INDEX_PATH>
```

–detect_input_dir Path to the directory containing BAM (and BAI files, see –index option). should be the same as –align_outdir.

–detect_outdir Directory where detection output will be saved.

`--genome_fasta` and `--genome_index_dir` are described above (in transformGenome and align, respectively).

2.3.4 align-detect

Executes *Pre-Analysis* and *Detection*. basic option:

```
./nextflow HE_scripts/MAIN_HE_SCRIPT.nf -c HE_scripts/MAIN_HE_SCRIPT.nf.config --
    run_mode align-detect --align_reads_dir <READS_FASTQ_PATH> --align_outdir <
    ALIGN_OUTPUT_PATH> --genome_fasta <PATH_TO_GENOME> --genome_index_dir <
    ORIGINAL_GENOME_INDEX_PATH> --transform_genome_dir <TRANSFORMED_GENOMES_DIR> --
    pair_end {0,1} --detect_outdir <DETECT_OUTPUT_PATH>
```

All options are already described above. Please note that `--detect_input_dir` is not required, as the pipeline integrates the output of *Pre-Analysis* into *Detection*.

3 Output Directories and Files

Output files are managed using the built-in `publishDir` option in Nextflow. Each part writes its output files to a designated directory specified by the corresponding `--*_outdir` options.

3.1 *Genome Transformation*

3.1.1 Transformed Genome Directory

The transformed genome files follow the naming convention:
$<*ref-base*>_<*alt-base*>_transformed_<GENOME_NAME>.fa/fasta.

3.1.2 Genome Indexing Directory

Subdirectories are created with names in the format: `${transformed_genome_file}_index`, containing all the index files generated by STAR.

3.2 *Pre-Analysis*

3.2.1 FASTP

The output files are generated and controlled by Fastp.

1. `.processed.fastp`: Fastq files with the successfully passed quality control reads.

2. `.json`: Summary output in JSON format, which contains information such as the number of reads that did not pass each filtering criterion.

3.2.2 First Map

The output files are generated and controlled by STAR.

1. `Aligned.out.bam`: Aligned BAM file from the first alignment (note that this file is not used for downstream analysis).

2. `Unmapped.out{.mate1/2}`: FASTQ files containing the sequences of unmapped reads from the first alignment.

3.2.3 Transform Unmapped

This directory contains subdirectories for each base combination, named *ref-base_alt-base*. Each subdirectory includes all the transformed unmapped FASTQ files for all samples in the run. The files are named using the format: *ref-base_alt-base*$sample_id.fastq.

3.2.4 Second Map

This directory contains subdirectories for each base combination, named *ref-base_alt-base*. Each subdirectory includes the aligned BAM files of the transformed reads aligned to the transformed genome. The files are named using the format `Aligned.out.bam`.

3.2.5 Re-Transform

This directory contains subdirectories for each base combination, named *ref-base_alt-base*. Each subdirectory includes the "re-transformed" aligned BAM files from the `Second Map` directory. These BAM files, prefixed with `.bam`, contain the original sequences restored from the FASTQ files in the `Transformed Unmapped` directory.

3.3 *Detection*

3.3.1 Detected Clusters

This directory contains subdirectories for each base combination, named *ref-base_alt-base*. Each subdirectory includes, for each sample:

1. `detected.csv`: A CSV file that parses each read in the input BAM files. The sequences of each record are compared to the genome provided in the `--genome_fasta` option, and information about the alignment is recorded.

Detectes CSV Attributes: Attributes can be output in two ways: 'all' or 'basic', controlled by the –detection_columns_select option (default: 'all').

Basic options:

Read_ID Unique identifier for each read.

Mate Indicates the mate of the read in paired-end data (always 1 in single-end data).

Chromosome The chromosome on which the read is aligned.

Strand The DNA strand (plus or minus) on which the read is aligned.

Position_0based The 0-based position of the read on the chromosome.

Alignment_length The length of the read alignment.

Read_Sequence The nucleotide sequence of the read.

Visualize_Alignment A visualization of the alignment (—: perfect match, *: editing site, X: mismatch) (not included in basic).

Reference_Sequence The reference sequence from the genome that aligns with the read.

Cigar The CIGAR string describing the alignment.

Flag The SAM flag indicating read properties.

Genomic_Position_Splicing_Blocks_0based Positions of splicing blocks on the genome in 0-based coordinates.

Read_Relative_Splicing_Blocks_0based Positions of splicing blocks relative to the read in 0-based coordinates.

Number_of_Editing_Sites Count of editing sites within the read.

Number_of_total_MM Total number of mismatches in the read alignment (including editing sites).

All options:

EditingSites_to_PhredScore_Map Mapping of editing sites to Phred quality scores.

MM_to_PhredScore_Map Mapping of mismatches to Phred quality scores.

A2C_MM List of A-to-C mismatches positions.

A2G_MM List of A-to-G mismatches positions.

A2T_MM List of A-to-T mismatches positions.

C2A_MM List of C-to-A mismatches positions.

C2G_MM List of C-to-G mismatches positions.

C2T_MM List of C-to-T mismatches positions.

G2A_MM List of G-to-A mismatches positions.

G2C_MM List of G-to-C mismatches positions.

G2T_MM List of G-to-T mismatches positions.

T2A_MM List of T-to-A mismatches positions.

T2C_MM List of T-to-C mismatches positions.

T2G_MM List of T-to-G mismatches positions.

Ref2N_MM List of reference-to-N mismatches positions.

NtoAlt_MM List of N-to-alternate mismatches positions.

3.3.2 Filtered Clusters

This directory contains subdirectories for each base combination, named *`ref-base_alt-base`*. Each subdirectory includes, for each sample, one or more of the following files. The specific files included can be controlled using the –filter_output_types option, which accepts the following values: `"all"`, `"passed"`, `"analysis"`, `"motifs"`, `"bed"`, and `"summary"`. The default value is `"all"`.

1. `passed.csv`: Contains all the metadata provided in the initial detection CSV file, along with additional information post-filtering, such as the number of passed editing sites (ES) and mismatches (MM).
2. `condition_analysis.csv`: This file provides a True/False assessment for each condition (passed/not passed) applied to each read.
3. `motifs.csv`: For each read, this file includes the count of each upstream base present in the total editing events (e.g., 3 of the events had T before them) and similarly for downstream bases. This information is used for later analysis of motifs related to the editing.
4. `clusters.bed`: Contains the genomic positions of each hyper-editing cluster.
5. `summary.json`: Provides statistics and summary information for the entire sample, such as the total number of passed reads and the average number of editing sites (ES).

Passed CSV Attributes: The `passed.csv` file contains all the attributes from the `detected.csv` file, with the addition of the following:

Number_of_Passed_ES Number of editing sites that pass the minimal Phred score threshold.

Number_of_Passed_MM Number of mismatches that pass the minimal Phred score threshold.

Editing_Fraction_Passed Editing fraction calculated as the number of passed editing sites divided by the number of total passed mismatches.

Passed_ES_Pos_to_Phred_Score_Map Mapping of passed editing sites to Phred quality scores, replacing the `ES_Pos_to_Phred_Score_Map`.

Average_ES_Phred_Score Average Phred score of all editing sites.

Average_Adjacent_ES_Distance Average distance between each pair of adjacent editing sites.

Condition Analysis CSV Attributes: For each condition, the value of the condition is appended as a suffix to the attribute title. The attributes in the `condition_analysis.csv` file include:

Read_ID Identifier for each read.

Passed_All Indicator of whether the read passed all conditions.

Edited Indicator of whether the read was edited (i.e., had a non-zero number of editing sites).

Min_Editing_Sites Indicator for the condition specifying the minimum number of editing sites required.

Min_Editing_to_Total_MM_Fraction Indicator for the condition specifying the minimum fraction of editing sites to total mismatches.

Min_Editing_Phred_Score Indicator for the condition specifying the minimum Phred score required for editing sites.

Min_Editing_to_Read_Length_Ratio Indicator for the condition specifying the minimum ratio of editing sites to read length.

Min_Cluster_Length_to_Read_Length_Ratio Indicator for the condition specifying the minimum ratio of cluster length to read length.

Motifs CSV Attributes: Includes the following attributes for each read, detailing the counts of upstream and downstream base appearances:

Read_ID Identifier for each read.

Mate Indicates the mate of the read in paired-end data (always 1 in single-end data).

upstream_A Count of adenine (A) bases upstream of the editing site.

upstream_C Count of cytosine (C) bases upstream of the editing site.

upstream_G Count of guanine (G) bases upstream of the editing site.

upstream_T Count of thymine (T) bases upstream of the editing site.

downstream_A Count of adenine (A) bases downstream of the editing site.

downstream_C Count of cytosine (C) bases downstream of the editing site.

downstream_G Count of guanine (G) bases downstream of the editing site.

downstream_T Count of thymine (T) bases downstream of the editing site.

Clusters BED Attributes: The `clusters.bed` file provides the genomic positions of each hyper-editing cluster. The attributes are as follows:

Column 1 Chromosome: The chromosome where the cluster is located.

Column 2 Cluster genomic start position.

Column 3 Cluster genomic end position.

Column 4 Name: The Read ID from which the cluster originated.

Column 5 Score: The number of editing sites within the cluster.

Column 6 Strand: The DNA strand orientation, represented as '+' or '-'.

4 Internal Scripts & Docker Dependencies

4.1 Internal Scripts Description

Each part of the workflow is implemented as a separate Nextflow script: *Genome Transformation* is implemented in `genome_setup.nf`, *Pre-Analysis* in `pre_analysis.nf`, and *Detection* in `HE_detection.nf`. Each script is accompanied by a corresponding configuration file. Each part can be invoked separately, using its respective configuration file, without the need to run the main script.

Each part is composed of processes, and any process that relies on external programs (beyond basic Bash commands) operates within a Docker container.

4.2 Default Docker Containers

4.2.1 genome_setup.nf

1. TRANSFORM: 'bashell/alpine-bash'
2. INDEX: 'quay.io/biocontainers/star:2.7.10b–h9ee0642_0'

4.2.2 pre_analysis.nf

1. FASTP: 'quay.io/biocontainers/fastp:0.23.4–hadf994f_2'
2. FIRST_STAR_MAP: 'quay.io/biocontainers/star:2.7.10b–h9ee0642_0'
3. SECOND_STAR_MAP: 'quay.io/biocontainers/star:2.7.10b–h9ee0642_0'
4. RETRANSFORM: 'quay.io/biocontainers/pysam:0.22.0–py38h15b938a_0'
5. INDEX_BAM: 'quay.io/biocontainers/samtools:1.20–h50ea8bc_1'

4.2.3 HE_detection.nf

1. INDEX_BAM: 'quay.io/biocontainers/samtools:1.20–h50ea8bc_1'
2. COUNT_RECORDS: 'quay.io/biocontainers/samtools:1.20–h50ea8bc_1'
3. DETECT: 'quay.io/biocontainers/pysam:0.22.0–py38h15b938a_0'
4. FILTER: 'quay.io/jupyter/scipy-notebook'

5 All Options Description

5.1 Main Script

5.1.1 Run Modes

`--run_mode` Specifies the mode of operation. Possible values are `transformGenome`, `align`, `detect`, and `align-detect`.

5.1.2 Genome Options

Can also be accessed in a separated run of each part.

`--genome_fasta` Path to the original genome FASTA file, which will be used either in the transformation of the genome or in the detection part (to compare the read to the alignment). The file must be in FASTA format with a `.fa` or `.fasta` suffix.

`--genome_index_dir` Path to the original genome index directory used for the first alignment. The index is used either in the transformation of the genome or in the detection part.

5.1.3 Files Options

Can also be accessed in an separated run of *Pre-Analysis* and *Detection*.

`--file_separator` Separator character used in file names.(default: "_")

`--mate_separator` Separator character between mates in file names. (default: "_")

`--suffix_separator` Separator character for suffixes in file names. (default: ".")

`--mate1_suff` Suffix for the first mate in paired-end reads. (default: 1)

`--mate2_suff` Suffix for the second mate in paired-end reads. (default: 2)

`--python_command` Command to run a Python script. (defualt: python).

5.2 *Genome Transformation*

`--genome_setup_outdir` Path to the directory where the *Genome Transformation* output will be stored.

`--transform_genome_output_dir` Path to the directory where the transformed genome files will be saved. Default is set to `$genome_setup_outdir/transformed_genome`.

`--index_output_dir` Path to the directory where the genome index files will be stored. Default is set to `$params.genome_setup_outdir/genome_index`.

5.3 *Pre-Analysis*

5.3.1 File Parameters

`--pair_end` Specifies whether the reads are paired-end (PE) or single-end (SE). Values are `0` for SE and `1` for PE.

`--align_reads_dir` Path to the directory containing the read files to be aligned.

`--reads_suffix` Suffix of the read files. (default: fastq)

`--SE_pattern` Pattern for single-end reads. (default: "$params.reads_suffix")

`--SE_reads` Path pattern for single-end read files. (default:"$params.align_reads_dir/*$params.SE_pattern")

`--PE_reads` Path pattern for paired-end read files. (default: "$params.align_reads_dir/*$params.mate_seperator$params.mate1_suff,$params.mate2_suff$params.reads_suffix")

5.3.2 Output Parameters

`--align_outdir` Path to the directory where alignment outputs will be stored.

`--fastp_output_dir` Directory for output files from the fastp preprocessing step. (default: `$params.align_outdir/fastp`.)

`--first_map_output_dir` Directory for the output of the first alignment step. (default: `$params.align_outdir/first_map`.)

`--transform_output_dir` Directory for the output of the transformed unmapped reads. (default:`$params.align_outdir/transformed_unmapped`.)

`--second_map_output_dir` Directory for the output of the second alignment step. (default: `$params.align_outdir/second_map`.)

`--retransform_output_dir` Directory for the output of the re-transformed reads. (default: `$params.align_outdir/re-transform`.)

5.3.3 FASTP Parameters

`--fastp_command` Command for the fastp preprocessing tool. (default: fastp)

`--N_bases_num` Maximum number of N bases allowed in a read. FASTP comand: -n. (default: 5)

`--avg_quality` Minimum average quality score of a read. FASTP comand: -e. (default: 30)

`--low_quality_per` Minimum percentage of bases allowed to be unqualified. FASTP comand: -u. (default: 20)

`--low_quality_num` Minimum quality value that a base must have to be considered qualified (Phred score). FASTP comand: -q. (default: 25)

`--complexity_threshold` Complexity threshold for filtering reads. FASTP comand: –complexity_threshold. (default: 30)

5.3.4 First STAR Map Parameters

`--fastp_output_suffix` Suffix for fastp processed read files. (default: ".processed.fastq")

`--fastq_reads` Path pattern for fastp processed read files. (default: "$params.fastp_output_dir/$params.mate_seperator$params.mate1_suff,$params.mate2_suff$params.fastp_output_suffix")

`--STAR_command` Command for the STAR aligner. (default: STAR)

`--STAR_MAX_PARALLEL` Maximum number of internal parallel STAR jobs. (default: 6)

`--read_files_command` Command line to execute for each input file. Corresponds to STAR's –readFilesCommand. (default: cat)

`--SAM_attr` Desired SAM attributes to include in the output. Corresponds to STAR's –outSAMattributes. (default: All)

`--outSAMtype` Output format for STAR aligner. Corresponds to STAR's –outSAMtype. (default: BAM Unsorted)

`--min_SJ_overhang` Minimum overhang for spliced alignments. Corresponds to STAR's –alignSJoverhangMin. (default: 8)

`--max_intron_size` Maximum intron length. Corresponds to STAR's –alignIntronMax. (default: 1000000)

`--max_mates_gap` Maximum genomic distance between mates. Corresponds to STAR's –alignMatesGapMax. (default: 600000)

`--max_mismatches_ratio_to_ref` Maximum ratio of mismatches to mapped length. Corresponds to STAR's –outFilterMismatchNoverLmax. (default: 0.3)

`--max_mismatches_ratio_to_read` Maximum ratio of mismatches to read length. Corresponds to STAR's –outFilterMismatchNoverReadLmax. (default: 1)

`--norm_num_of_matches` Minimum number of matched bases normalized to the read length. Corresponds to STAR's –outFilterMatchNminOverLread. (default: 0.66)

`--max_num_of_allignment_first_map` Maximum number of multiple alignments allowed for a read. Corresponds to STAR's –outFilterMultimapNmax. (default: 5)

`--genome_load_set` Setting for genome shared memory usage. Corresponds to STAR's –genomeLoad. (default: NoSharedMemory)

`--num_of_threads` Number of threads to use. Corresponds to STAR's –runThreadN. (default: 5)

`--unmapped_out_files` Output files for unmapped reads. Corresponds to STAR's –outReadsUnmapped. (default: Fastx)

`--output_files_permissions` Permissions for output files. Corresponds to STAR's –runDirPerm. (default: All_RWX)

5.3.5 Transform Reads Parameters

5.3.6 Second STAR Map Parameters

`--transform_genome_dir` Directory for the transformed genome. required for `align` and `align-detect` run modes. Should corresponds to `--genome_setup_outdir` option.

`--transformed_indexes` Path pattern for transformed genome index files, required for `align` and `align-detect` run modes. Should corresponds to `--index_output_dir` option.
(default: "$params.transform_genome_dir/genome_index/*")

`--max_num_of_allignment_second_map` Maximum number of alignments for second mapping. Corresponds to STAR's –outFilterMultimapNmax. (default: 20)

`--second_map_genome_load_set` Settings for genome loading during second mapping. Corresponds to STAR's –genomeLoad. (default: LoadAndKeep)

5.3.7 Retransform Reads Parameters

`--STAR_unmapped_suffix` Suffix for unmapped reads from STAR. (default: .Unmapped.out.mate)

`--filter_sam_files` Pattern for filtering SAM files. (default: '*Aligned.out*')

`--retransform_python_script` Path to the re-transform Python script.

5.3.8 Index Parameters

`--index_threads` Number of threads for indexing. (default: 16)

5.4 *Detection*

5.4.1 Input Parameters

`--bam_suffix` Suffix for input BAM files. (default: ".bam")

`--bai_suffix` Suffix for input BAI (BAM index) files. (default: ".bai")

`--detect_input_dir` Directory for *Detection* part input. Can be either the output directory of *Pre-Analysis*, or a directory containing input BAM and/or BAI files (for independent run). Required.

`--retransform_dir_name` Name of the directory where the output files of the *Pre-Analysis* (re-transformed BAM files) are stored. Not used in an independent run. (default: "re-transform")

`--HE_reads` Path pattern for hyper-edited (HE) reads for a regular run (dependent on *Pre-Analysis* output). (default: "$params.detect_input_dir/
$params.retransform_dir_name/**/*$params.bam_suffix,$params.bai_suffix}")

`--HE_reads_independent` Path pattern for HE reads in an independent run.
(default: "$params.detect_input_dir/**{$params.bam_suffix,$params.bai_suffix}")

`--detect_python_script` Path to the Python script for parallel detection.
(default: "/private10/Projects/Gili/HE_workdir/HE_scripts/parallel_detection.py")

`--filter_python_script` Path to the Python script for parallel filtering.
(default: "/private10/Projects/Gili/HE_workdir/HE_scripts/parallel_filter.py")

`--PE_filter_python_script` Path to the Python script for paired-end filtering.

Please note that the `**` option in Nextflow enables recursive pattern matching, allowing the search to extend into subdirectories as well.

5.4.2 Independent Run

Detection's independent mode specifies that the BAM files are not derived from *Pre-Analysis*. In this mode, the BAM files are compared to the reference genome to detect hyper-editing across the 12 possible base combinations. If the files are not indexed, you should enable the `--index` flag.

`--independent` Boolean flag indicating whether the run is independent. (default: `false`)

`--index_threads` Number of threads to use for indexing. (default: `16`)

`--index` Boolean flag indicating whether indexing is enabled. (default: `false`)

5.4.3 Output Parameters

`--detect_outdir` Directory for storing detection outputs. Required.

`--index_output_dir` Directory for storing index files, typically under `detect_outdir`.
(default: "$params.detect_outdir/index")

`--detect_output_dir` Directory for storing the detected clusters files.
(default: "$params.detect_outdir/detected_clusters")

`--filter_output_dir` Directory for storing filtered clusters files.
(default: "$params.detect_outdir/filtered_clusters")

5.4.4 Detection Script Parameters

`--detection_columns_select` Specifies which columns to include in the detection output. Options are `'all'` or `'basic'`. (default: `'all'`)

`--max_detection_threads` Maximum number of threads to use for internal parallelism of the detection script. (default: `3`)

`--detection_batch_size` Size of batches for detection processing. If set to `0`, the size is automatically determined using number of records in the BAM files. (default: `0`)

5.4.5 Filter Script Parameters

`--filter_output_types` Specifies the types of output for filtering. Options are `"all"`, `"passed"`, `"analysis"`, `"motifs"`, `"bed"`, and `"summary"`. (default: `'all'`)

`--max_filter_threads` Maximum number of threads to use for internal parallelism of the filtering script. (default: `3`)

`--filter_batch_size` Size of batches for filtering processing. If set to `0`, the size is automatically determined using number of records in the detection output files. (default: `0`)

`--min_editing_sites` Minimum number of editing sites required for a read to be considered edited. (default: `1`)

`--min_editing_fraction` Minimum fraction of editing sites relative to total sites required for a read to be considered hyper-edited. (default: `0.6`)

`--min_phred_score` Minimum Phred score required for a base to be considered. (default: `30`)

`--min_es_length_ratio` Minimum ratio of the length of editing sites to the total length of the read. (default: `0.05`)

`--min_cluster_length_ratio` Minimum ratio of the length of the cluster to the total length of the read. (default: `0.1`)