

Package ‘Rbowtie2’

October 16, 2018

Type Package

Title An R Wrapper for Bowtie2 and AdapterRemoval

Version 1.2.0

Date 2017-08-18

Author Zheng Wei, Wei Zhang

Maintainer Zheng Wei <wzweizheng@qq.com>

Description

This package provides an R wrapper of the popular bowtie2 sequencing reads aligner and AdapterRemoval, a convenient tool for rapid adapter trimming, identification, and read merging.

License GPL (>= 3)

Depends R (>= 3.5)

Suggests knitr

SystemRequirements C++11

Archs x64

RoxygenNote 6.0.1

biocViews Sequencing, Alignment, Preprocessing

VignetteBuilder knitr

git_url <https://git.bioconductor.org/packages/Rbowtie2>

git_branch RELEASE_3_7

git_last_commit b37ba41

git_last_commit_date 2018-04-30

Date/Publication 2018-10-15

R topics documented:

adapterremoval_usage	2
adapterremoval_version	2
bowtie2	3
bowtie2-build	4
bowtie2_build_usage	5
bowtie2_usage	6
bowtie2_version	7
identify_adapters	7
remove_adapters	8

Index**11**

adapterremoval_usage *Print available arguments for adapterremoval*

Description

Print available arguments for adapterremoval. Note that some arguments to the adapterremoval are invalid if they are already handled as explicit function arguments.

Usage

```
adapterremoval_usage()
```

Value

AdapterRemoval available arguments and their usage.

Author(s)

Zheng Wei

References

Schubert, Lindgreen, and Orlando (2016). AdapterRemoval v2: rapid adapter trimming, identification, and read merging. BMC Research Notes, 12;9(1):88.

Examples

```
adapterremoval_usage()
```

adapterremoval_version

Print version information of adapterremoval

Description

Print version information of adapterremoval

Usage

```
adapterremoval_version()
```

Value

An invisible Integer of call status. The value is 0 when there is not any mistakes

Author(s)

Zheng Wei

References

Schubert, Lindgreen, and Orlando (2016). AdapterRemoval v2: rapid adapter trimming, identification, and read merging. BMC Research Notes, 12;9(1):88.

Examples

```
adapterremoval_version()
```

bowtie2 *Interface to bowtie2 of bowtie2-2.2.3*

Description

This function can be use to call wrapped bowtie2 binary.

Usage

```
bowtie2(bt2Index, samOutput, seq1, ..., seq2 = NULL, interleaved = FALSE,
        overwrite = FALSE)
```

Arguments

bt2Index	Character scalar. bowtie2 index files prefix: 'dir/basename' (minus trailing '.*.bt2' of 'dir/basename.*.bt2').
samOutput	Character scalar. A path to a SAM file used for the alignment output.
seq1	Character vector. For single-end sequencing, it contains sequence file paths. For paired-end sequencing, it can be file paths with #1 mates paired with file paths in seq2. And it can also be interleaved file paths when argument interleaved=TRUE
...	Additional arguments to be passed on to the binaries. See below for details.
seq2	Character vector. It contains file paths with #2 mates paired with file paths in seq1. For single-end sequencing files and interleaved paired-end sequencing files(argument interleaved=TRUE), it must be NULL.
interleaved	Logical. Set TRUE when files are interleaved paired-end sequencing data.
overwrite	Logical. Force overwriting of existing files if setting TRUE.

Details

All additional arguments in ... are interpreted as additional parameters to be passed on to bowtie2. All of them should be Character or Numeric scalar. You can put all additional arguments in one Character(e.g. "-threads 8 -no-mixed") with white space splited just like command line, or put them in different Character (e.g. "-threads", "8", "-no-mixed"). Note that some arguments("-x", "-interleaved", "-U", "-1", "-2", "-S") to the bowtie2 are invalid if they are already handled as explicit function arguments. See the output of bowtie2_usage() for details about available parameters.

Value

An invisible Integer of call status. The value is 0 when there is not any mistakes Otherwise the value is non-zero.

Author(s)

Zheng Wei

References

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature methods*, 9(4), 357-359.

Examples

```
td <- tempdir()
## Building a bowtie2 index
refs <- dir(system.file(package="Rbowtie2", "extdata", "bt2","refs"),
full=TRUE)
bowtie2_build(references=refs, bt2Index=file.path(td, "lambda_virus"),
"--threads 4 --quiet",overwrite=TRUE)
## Alignments
reads_1 <- system.file(package="Rbowtie2", "extdata", "bt2", "reads",
"reads_1.fastq")
reads_2 <- system.file(package="Rbowtie2", "extdata", "bt2", "reads",
"reads_2.fastq")
if(file.exists(file.path(td, "lambda_virus.1.bt2"))){
  cmdout<-bowtie2(bt2Index = file.path(td, "lambda_virus"),
  samOutput = file.path(td, "result.sam"),
  seq1=reads_1,seq2=reads_2,overwrite=TRUE,"--threads 3");cmdout
  head(readLines(file.path(td, "result.sam")))
}
```

 bowtie2-build

Interface to bowtie2-build of bowtie2-2.2.3

Description

This function can be use to call wrapped bowtie2-build binary

Usage

```
bowtie2_build(references, bt2Index, ..., overwrite = FALSE)
```

Arguments

references	Character vector. The path to the files containing the references for which to build a bowtie index.
bt2Index	Character scalar. Write bowtie2 index data to files with this prefix: 'dir/basename'. If the files with path like 'dir/basename.*.bt2' already exists, the function function will cast an error, unless argument overwrite is TRUE.
...	Additional arguments to be passed on to the binaries. See below for details.
overwrite	Logical. Force overwriting of existing files if setting TRUE.

Details

All additional arguments in ... are interpreted as additional parameters to be passed on to bowtie2_build. All of them should be Character or Numeric scalar. You can put all additional arguments in one Character(e.g. "--threads 8 --quiet") with white space splited just like command line, or put them in different Character(e.g. "--threads","8","--quiet"). See the output of bowtie2_build_usage() for details about available parameters.

Value

An invisible Integer of call status. The value is 0 when there is not any mistakes Otherwise the value is non-zero.

Author(s)

Zheng Wei

References

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. Nature methods, 9(4), 357-359.

Examples

```
td <- tempdir()
## Building a bowtie2 index
refs <- dir(system.file(package="Rbowtie2", "extdata", "bt2", "refs"),
full=TRUE)
cmdout<-bowtie2_build(references=refs, bt2Index=file.path(td, "lambda_virus"),
"--threads 4 --quiet",overwrite=TRUE);cmdout
## Use additional arguments in another way
cmdout<-bowtie2_build(references=refs, bt2Index=file.path(td, "lambda_virus"),
"--threads",4,"--quiet",overwrite=TRUE);cmdout
## The function will print the output
## during the process without "--quiet" argument.
cmdout<-bowtie2_build(references=refs, bt2Index=file.path(td, "lambda_virus"),
overwrite=TRUE);cmdout
```

bowtie2_build_usage *Print available arguments for bowtie2_build_usage*

Description

Note that some arguments to the bowtie2_build_usage are invalid if they are already handled as explicit function arguments.

Usage

```
bowtie2_build_usage()
```

Value

bowtie2_build available arguments and their usage.

Author(s)

Zheng Wei

References

Langmead B, Salzberg S. Fast gapped-read alignment with Bowtie 2. Nature Methods. 2012, 9:357-359.

Examples

```
bowtie2_build_usage()
```

bowtie2_usage

Print available arguments for bowtie2

Description

Note that some arguments to the bowtie2 are invalid if they are already handled as explicit function arguments.

Usage

```
bowtie2_usage()
```

Value

bowtie2 available arguments and their usage.

Author(s)

Zheng Wei

References

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. Nature methods, 9(4), 357-359.

Examples

```
bowtie2_usage()
```

bowtie2_version *Print version information of bowtie2-2.2.3*

Description

Print version information of bowtie2-2.2.3

Usage

```
bowtie2_version()
```

Value

An invisible Integer of call status. The value is 0 when there is not any mistakes Otherwise the value is non-zero.

Author(s)

Zheng Wei

References

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with Bowtie 2. *Nature methods*, 9(4), 357-359.

Examples

```
cmdout<-bowtie2_version();cmdout
```

identify_adapters *identify adapters for paired-end reads*

Description

This function can be use to call wrapped AdapterRemoval binary for adapters identifying.

Usage

```
identify_adapters(file1, file2, ..., basename = NULL, overwrite = FALSE)
```

Arguments

file1	Character vector. It can be file paths with #1 mates paired with file paths in file2 And it can also be interleaved file paths when argument interleaved=TRUE
file2	Character vector. It contains file paths with #2 mates paired with file paths in file1. For interleaved paired-end sequencing files(argument interleaved=TRUE),it must to be setted to NULL.
...	Additional arguments to be passed on to the binaries. See below for details.
basename	Character. The outputfile path prefix. Default: your_output
overwrite	Logical. Force overwriting of existing files if setting TRUE.

Details

All additional arguments in ... are interpreted as additional parameters to be passed on to identify_adapters. All of them should be Character or Numeric scalar. You can put all additional arguments in one Character (e.g. "--threads 8") with white space splited just like command line, or put them in different Character (e.g. "--threads", "8"). Note that some arguments ("--identify-adapters", "--file1", "--file2", "--basename") to the identify_adapters are invalid if they are already handled as explicit function arguments. See the output of adapterremoval_usage() for details about available parameters.

Value

An invisible Character vector of adapters for each mate.

Author(s)

Zheng Wei

References

Schubert, Lindgreen, and Orlando (2016). AdapterRemoval v2: rapid adapter trimming, identification, and read merging. BMC Research Notes, 12;9(1):88.

Examples

```
td <- tempdir()
reads_1 <- system.file(package="Rbowtie2", "extdata", "adrm", "reads_1.fq")
reads_2 <- system.file(package="Rbowtie2", "extdata", "adrm", "reads_2.fq")
adapters <- identify_adapters(file1=reads_1, file2=reads_2,
  basename = file.path(td, "reads")
  , "--threads 2", overwrite=TRUE)
adapters
```

remove_adapters

Interface to bowtie2 of adapterremoval-2.2.1a

Description

This function can be use to call wrapped AdapterRemoval binary.

Usage

```
remove_adapters(file1, ..., adapter1 = NULL, output1 = NULL, file2 = NULL,
  adapter2 = NULL, output2 = NULL, basename = NULL, interleaved = FALSE,
  overwrite = FALSE)
```

Arguments

file1 Character vector. For single-end sequencing, it contains sequence file paths. For paired-end sequencing, it can be file paths with #1 mates paired with file paths in file2 And it can also be interleaved file paths when argument interleaved=TRUE

... Additional arguments to be passed on to the binaries. See below for details.

adapter1	Character. It is an adapter sequence for file1. Default: AGATCGGAAGAG-CACACGTCTGAACTCCAGTCACNNNNNNATCTCGTATGCCGTCTTCTGCTTG
output1	Character. The trimmed mate1 reads output file path for file1. Default: base-name.pair1.truncated (paired-end), basename.truncated (single-end), or base-name.paired.truncated (interleaved)
file2	Character vector. It contains file paths with #2 mates paired with file paths in file1. For single-end sequencing files and interleaved paired-end sequencing files(argument interleaved=TRUE), it must be NULL.
adapter2	Character. It is an adapter sequence for file2. Default: AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT
output2	Character. The trimmed mate2 reads output file path for file2. Default: BASE-NAME.pair2.truncated (only used in PE mode, but not if <code>--interleaved-output</code> is enabled)
basename	Character. The outputfile path prefix. Default: your_output
interleaved	Logical. Set TRUE when files are interleaved paired-end sequencing data.
overwrite	Logical. Force overwriting of existing files if setting TRUE.

Details

All additional arguments in ... are interpreted as additional parameters to be passed on to `remove_adapters`. All of them should be `Character` or `Numeric` scalar. You can put all additional arguments in one `Character`(e.g. `"--threads 8"`) with white space splited just like command line, or put them in different `Character`(e.g. `"--threads", "8"`). Note that some arguments(`"--file1", "--file2", "--adapter1", "--adapter2", "--output1", "--output2", "--basename", "--interleaved"`) to the `identify_adapters` are invalid if they are already handled as explicit function arguments. See the output of `adapterremoval_usage()` for details about available parameters.

Value

An invisible `Integer` of call status. The value is 0 when there is not any mistake. Otherwise the value is non-zero.

Author(s)

Zheng Wei

References

Schubert, Lindgreen, and Orlando (2016). AdapterRemoval v2: rapid adapter trimming, identification, and read merging. *BMC Research Notes*, 12;9(1):88.

Examples

```
td <- tempdir()

# Identify adapters
reads_1 <- system.file(package="Rbowtie2", "extdata", "adrm", "reads_1.fq")
reads_2 <- system.file(package="Rbowtie2", "extdata", "adrm", "reads_2.fq")
adapters <- identify_adapters(file1=reads_1,file2=reads_2,
  basename=file.path(td,"reads"), "--threads 3",overwrite=TRUE)

# Remove adapters
```

```
cmdout<-remove_adapters(file1=reads_1,file2=reads_2,adapter1 = adapters[1],  
adapter2 = adapters[2],  
output1=file.path(td,"reads_1.trimmed.fq"),  
output2=file.path(td,"reads_2.trimmed.fq"),  
basename=file.path(td,"reads.base"),overwrite=TRUE,"--threads 3");cmdout
```

Index

`adapterremoval_usage`, [2](#)
`adapterremoval_version`, [2](#)

`bowtie2`, [3](#)
`bowtie2-build`, [4](#)
`bowtie2_build` (`bowtie2-build`), [4](#)
`bowtie2_build_usage`, [5](#)
`bowtie2_usage`, [6](#)
`bowtie2_version`, [7](#)

`identify_adapters`, [7](#)

`remove_adapters`, [8](#)