

Package ‘rstoat’

November 10, 2021

Version 1.0.2

Date 2021-10-21

Title R Interface to the 'STOAT' Platform

Author John Wilshire [aut, cre],
Richard Li [aut],
Ajay Ranipeta [aut]

Maintainer John Wilshire <john.wilshire@yale.edu>

Description

A wrapper for the 'Spatiotemporal Observation Annotation Tool' ('STOAT', <<https://www.mol.org/stoat>>) which allows users to run annotation jobs and retrieve results in the R environment.

License BSD_3_clause + file LICENSE

URL <https://www.mol.org/stoat>

BugReports <https://github.com/MapofLife/rstoat/issues>

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

Imports htr, jsonlite, curl, keyring

Suggests knitr, rstudioapi, dplyr, rmarkdown

VignetteBuilder knitr, rmarkdown

RoxygenNote 7.1.2

NeedsCompilation no

Repository CRAN

Date/Publication 2021-11-10 05:10:02 UTC

R topics documented:

download_annotation	2
download_sample_data	3
get_products	3

job_details	4
job_species	4
melopsittacus_short	5
mol_login	5
my_datasets	6
my_jobs	6
ninox_short	7
read_output	7
start_annotation_batch	8
start_annotation_simple	9

Index	12
--------------	-----------

download_annotation	<i>Download annotation results</i>
---------------------	------------------------------------

Description

Download results of a successfully completed batch annotation. Requires login, please run mol_login(<email_address>)
 Uses the output from my_jobs() for the annotation id.

Usage

```
download_annotation(annotation_id, dir = "annotation_results")
```

Arguments

annotation_id	The id of the annotation
dir	The directory where to write the annotation.

Value

The path of the downloaded annotation.

Examples

```
## Not run:
download_annotation(<annotation_id>, <dir>)

## End(Not run)
```

download_sample_data *Download sample annotation data*

Description

Download the powerful owl and budgerigar sample datasets (both raw occurrence data and annotated data), used in the Introduction vignette, from Map of Life's datastore.

Usage

```
download_sample_data(dir = "sample_data")
```

Arguments

dir The directory where to store the data.

Value

The path of the downloaded sample data.

Examples

```
## Not run:  
download_sample_data()  
  
## End(Not run)
```

get_products *Retrieve product metadata*

Description

Get information on available products for annotation. Get the spatial and temporal buffer limits for use in when creating a custom annotation.

Usage

```
get_products()
```

Value

A data.frame of spatial and temporal buffer limits

Examples

```
## Not run:  
get_products()  
  
## End(Not run)
```

job_details	<i>Retrieve annotation job details</i>
-------------	--

Description

Get details of a batch annotation job. Requires login, please run `mol_login(<email_address>)`. Uses the output from `my_jobs()`.

Usage

```
job_details(annotation_id)
```

Arguments

`annotation_id` The annotation id from from `my_jobs()`.

Value

A data.frame of layers and their statuses, along with the `annotation_id`, and the `dataset_id` for the custom annotation.

Examples

```
## Not run:  
job_details(<annotation_id>)  
  
## End(Not run)
```

job_species	<i>View annotation job species</i>
-------------	------------------------------------

Description

View the species in a completed annotation and other details. Only works for successfully completed jobs. Requires login, please run `mol_login(<email_address>)` Uses the output from `my_jobs()`.

Usage

```
job_species(annotation_id)
```

Arguments

`annotation_id` The annotation id from from `my_jobs()`.

Value

A data.frame, with species and counts in this annotation.

Examples

```
## Not run:  
job_species(<annotation_id>  
  
## End(Not run)
```

melopsittacus_short *Short budgerigar sample dataset*

Description

Dataset containing 1500 annotated records of the budgerigar. Subsampled version of the dataset downloaded using `download_sample_data()`.

Usage

```
melopsittacus_short
```

Format

A data frame with 1500 rows and 10 variables:

Source

See introduction vignette

mol_login *Map of Life Login*

Description

Login to your Map of Life account.

Usage

```
mol_login(email, password = NULL)
```

Arguments

email	The email address associated with your Map of Life Account.
password	Your map of life password. If left blank, and you are in RStudio you can enter it via a secure popup.

Value

No return value

Examples

```
## Not run:  
mol_login("your.email@company.com")  
  
## End(Not run)
```

my_datasets	<i>View datasets</i>
-------------	----------------------

Description

List logged-in user's uploaded datasets (uploaded to <https://mol.org> through <https://mol.org/upload>). Requires login, please run `mol_login(<email_address>)`

Usage

```
my_datasets()
```

Value

A data frame of a users datasets and their associated ids.

Examples

```
## Not run:  
my_datasets()  
  
## End(Not run)
```

my_jobs	<i>List all jobs</i>
---------	----------------------

Description

List logged-in user's past and current annotation jobs. Requires login, please run `mol_login(<email_address>)`

Usage

```
my_jobs()
```

Value

A data.frame containing jobs metadata.

Examples

```
## Not run:  
my_jobs()  
  
## End(Not run)
```

ninox_short	<i>Short powerful owl sample dataset</i>
-------------	--

Description

Dataset containing 1500 annotated records of the powerful owl. Subsampled version of the dataset downloaded using `download_sample_data()`.

Usage

```
ninox_short
```

Format

A data frame with 1500 rows and 11 variables:

Source

See introduction vignette

read_output	<i>Read annotation output into R</i>
-------------	--------------------------------------

Description

Convenience function which reads and joins annotation results spread across multiple files for space efficiency. To run this function, please first download annotated data first using `download_annotation()`

Usage

```
read_output(directory, drop_event_id = TRUE)
```

Arguments

directory	The path of the data.
drop_event_id	Whether to drop the event_id column or not.

Value

A data.frame of annotated data, one row per variable per event

Examples

```
## Not run:
read_output("path/to/your/downloaded/data/directory")

## End(Not run)
```

start_annotation_batch

Start batch annotation

Description

Submit a dataset previously uploaded on mol.org for annotation. To upload a dataset please visit <https://mol.org/upload/> Requires login, please run mol_login(<email_address>)

Usage

```
start_annotation_batch(dataset_id, title, layers)
```

Arguments

dataset_id	The id of the dataset. List your datasets with the my_datasets() function.
title	The title of the annotation job.
layers	A list of parameters or vector of codes, of the layers, see the examples below.

Value

No return value, check my_jobs() to confirm successful job submission.

Examples

```
## Not run:
start_annotation_batch('<dataset_id>', 'My annotation task', layers = list(
  list(product = "chelsea", variable = "precip", spatial = 1000, temporal = 30)
))
# alternatively supplying the code is fine.
start_annotation_batch('<dataset_id>',
  'My 2nd annotation task', layers = c("modis-ndvi-1000-1", "modis-1st_day-1000-1"))

## End(Not run)
```

`start_annotation_simple`*Start simple annotation*

Description

Submit a dataframe for on-the-fly annotation. Does not require login - for use for small numbers of records and pilot jobs.

You can annotate using layers that are on earthengine! Layer parameters: Used by all:

- spatial: The spatial buffer in meters.
- temporal: The temporal buffer in days.

For STOAT layers:

- product: The product e.g "srtm", or "landsat8".
- variable: The variable, e.g "elevation", or "evi".

For non STOAT layers, ie those in Google Earth Engine:

- id: The id of the image in Google Earth Engine, mapped to "product" in the output.
- static: Whether to load the imagery as an ImageCollection or as an Image.
- bands: A list, with one element, which is used to specify which band of the imagery to use for the annotation, mapped to "variable" in the output.
- reducers: A list containing one or more of the following:
 - mean
 - lcv_count
 - mode
 - median
 - stdev
 - min
 - max
 - stdev

If only one is provided the output will be assigned to "value", else the output will be named the same as the reducer.

Usage

```
start_annotation_simple(  
  events,  
  layers,  
  coords = c("lng", "lat"),  
  date = "date"  
)
```

Arguments

events	A data.frame for on the fly annotation
layers	A list of parameters or vector of codes, of the layers, see the examples below.
coords	A vector of length 2 containing column names for record longitudes, and latitudes.
date	Column name for record dates, dates must take the format YYYY-MM-DD

Value

Input data.frame with values from the annotation appended, in addition to unique identifier field event_id.

- event_id: A unique identifier for each occurrence
- product: Product used for annotation
- variable: Variable used for annotation
- s_buff: Spatial buffer in meters applied to occurrence
- t_buff: Temporal buffer in days applied to occurrence
- value: Annotated value of occurrence from requested layer (mean within buffer), if there is only one reducer (default), then this value will be here.
- stdev: Standard deviation of values within buffer
- valid_pixel_count: Number of pixels within buffered area'

Examples

```
## Not run:
events <- data.frame(
  event_id = as.character(1:2),
  lng = c(-4, 24),
  lat = c(10, 10),
  date = '2015-01-01'
)

# simple layer string format: PRODUCT-VARIABLE-S_BUFF-T_BUFF
layers <- 'landsat8-evi-100-16'
start_annotation_simple(events, layers)

start_annotation_simple(events, layers)

# For lcv_count (Count of landcover value), 'value' returned is a string of
# landcover counts within the AOI.
# The output format is:
# <LANDCOVER_CLASS>:<COUNT_OF_PIXELS_WITH_THAT_CLASS>
# classes are separated by commas.

start_annotation_simple(events, list(
  list(
    id="COPERNICUS/Landcover/100m/Proba-V-C3/Global",
    s_buff=1000,
```

```
        reducers=list("lcv_count", "mode"),
        static=FALSE,
        t_buff=365,
        bands=list("discrete_classification")
    )
))
```

```
# Annotating with two worldclim layers:
# bio01 is annual mean temperature
# bio12 is annual precipitation
```

```
start_annotation_simple(events, list(
  list(
    "id"= "WORLDCLIM/V1/BIO",
    "s_buff"=1000,
    "reducers"=list("mean"),
    "static"= TRUE,
    "t_buff"= 1,
    "bands"=list("bio01")
  ),
  list(
    "id"= "WORLDCLIM/V1/BIO",
    "s_buff"=1000,
    "reducers"=list("mean"),
    "static"= TRUE,
    "t_buff"= 1,
    "bands"=list("bio12")
  )
))
```

```
## End(Not run)
```

Index

* datasets

- melopsittacus_short, 5
- ninox_short, 7

- download_annotation, 2
- download_sample_data, 3

- get_products, 3

- job_details, 4
- job_species, 4

- melopsittacus_short, 5
- mol_login, 5
- my_datasets, 6
- my_jobs, 6

- ninox_short, 7

- read_output, 7

- start_annotation_batch, 8
- start_annotation_simple, 9