Package ‘clarifai’

April 12, 2017

Title Access to Clarifai API
Version 0.4.2
Maintainer Gaurav Sood <gsood07@gmail.com>
Description Get description of images from Clarifai API. For more information, see <http://clarifai.com>. Clarifai uses a large deep learning cloud to come up with descriptive labels of the things in an image. It also provides how confident it is about each of the labels.

URL http://github.com/soodoku/clarifai
BugReports http://github.com/soodoku/clarifai/issues
Depends R (>= 3.2.0)
License MIT + file LICENSE
LazyData true
VignetteBuilder knitr
Imports curl, jsonlite
Suggests knitr, rmarkdown, testthat
RoxygenNote 6.0.1
NeedsCompilation no
Author Gaurav Sood [aut, cre]
Repository CRAN
Date/Publication 2017-04-12 13:41:15 UTC

R topics documented:

clarifai-package .................................................. 2
clarifai_check_auth ............................................. 2
clarifai_check_results ......................................... 3
clarifai_check_token ........................................... 3
clarifai_GET ..................................................... 3
clarifai_POST .................................................... 4
feedback .......................................................... 4
clarifai_check_auth

get_color .................................................. 5
get_color_urls ............................................. 6
get_info ..................................................... 7
get_langs ..................................................... 8
get_token ................................................... 8
get_usage .................................................... 9
secret_id .................................................... 10
tag_images .................................................. 11
tag_image_urls ............................................ 12

Index

clarifai-package     clarifai: Get Description of (Moving) Images

Description


Your need credentials to use this application. If you haven’t already, you can get this at https://developer.clarifai.com/.

Author(s)

Gaurav Sood

See Also

get_token

clarifai_check_auth     Check if authentication information is in the environment

Description

Check if authentication information is in the environment

Usage

clarifai_check_auth()
clarifai_check_results

Description
Check results

Usage
clarifai_check_results(res)

Arguments
res results being returned by the API

clarifai_check_token Check if authentication token is in the environment

Description
Check if authentication token is in the environment

Usage
clarifai_check_token()

clarifai_GET GET

Description
GET

Usage
clarifai_GET(path, query, ...)

Arguments
path path to specific API request URL
query query list
... Additional arguments passed to curl_fetch_memory.

Value
list
clarifai_POST

Base POST AND GET functions. Not exported.

Description

POST

Usage

clarifai_POST(path, query = NULL, ...)

Arguments

path: path to specific API request URL
query: query list
... Additional arguments passed to curl_fetch_memory.

Value

list

feedback

Provide Feedback on Tags of an Image

Description

Suggest tags, suggest removal of tags, suggest (dis)similar documents, or let clarifai know which tag was clicked on

Usage

feedback(file_path = "", feedback_type = "add_tags", feedback_value = "", ...

Arguments

file_path: image on which the feedback is being provided
feedback_type: must be one of the following types: add_tags vector of suggested tags; default remove_tags vector of tags you suggest be removed search_click which tag(s) where clicked on similar_docids similar documents dissimilar_docids dissimilar documents
feedback_value: what tags to add, remove etc.
... Additional arguments passed to clarifai_POST.
get_color

References

https://developer.clarifai.com/guide/

Examples

```r
## Not run:

# Before calling the function, set API secret and id via
# secret_id(c("client_id", "secret"))
# and get token via get_token()

feedback(file_path="path_to_image", feedback_type="add_tags", feedback_value="suggested_tag")

## End(Not run)
```

---

**get_color**  
*Get the dominant colors present in images or videos*

**Description**

Get the dominant colors present in images or videos

**Usage**

```r
get_color(file_paths = NULL, ...)
```

**Arguments**

- `file_paths`  
a vector of paths to image file(s) for which you want tags

- `...`  
Additional arguments passed to `clarifai_POST`.

**Value**

Named list of length 3: `status_code`, `status_msg`, and `results`. `results` is a named list of length 12. Contains information about max. and min. image and video size allowed etc.

Prints `status_msg` by default

**References**

https://developer.clarifai.com/
get_color_urls

get_color_urls is a function that returns the dominant colors present in images or videos. It takes a string or character vector of URLs of images as its primary argument (img_urls). It also accepts a boolean argument (meta) which toggles whether or not you want the entire object. Additional arguments are passed to the clarifai_POST function.

**Usage**

```
get_color_urls(img_urls = NULL, meta = FALSE, ...)
```

**Arguments**

- `img_urls`: A string or character vector of URLs of images for which you want tags.
- `meta`: Boolean that toggles whether or not you want the entire object. Default is FALSE. When FALSE, the function returns just a data.frame of results.
- `...`: Additional arguments passed to `clarifai.POST`.

**Value**

Named list of length 3: `status_code`, `status_msg`, and `results`. `results` is a named list of length 12. Contains information about max. and min. image and video size allowed etc. Prints `status_msg` by default.

**References**

https://developer.clarifai.com/

**Examples**

```r
## Not run:

# Before calling the function, set API secret and id via secret_id(c("client_id", "secret"))
# and get token via get_token()

get_color_urls("https://samples.clarifai.com/metro-north.jpg")
get_color_urls(c("https://samples.clarifai.com/metro-north.jpg",

"https://samples.clarifai.com/metro-north.jpg"))
```
**get_info**

## Description

Basic information about the application — what are maximum sizes allowed, default language, max. and min. image and video size, max. batch size, etc.

## Usage

```r
get_info(...)  
```

## Arguments

... Additional arguments passed to `clarifai_GET`.

## Value

Named list of length 3: `status_code`, `status_msg`, and `results`. `results` is a named list of length 12. Contains information about max. and min. image and video size allowed etc.

Prints `status_msg` by default

## References

https://developer.clarifai.com/

## Examples

```r
## Not run:

# Before calling the function, set API secret and id via `secret_id(c("client_id", "secret"))`
# and get token via `get_token()`

get_info()

## End(Not run)
```
get_langs  List all the languages that the tag API call supports

Description
List all the languages that the tag API call supports

Usage
get_langs(...)

Arguments
...  Additional arguments passed to clarifai_GET.

Value
character vector listing all the languages that tag API call supports. See tag_images and codetag_image_urls

References
https://developer.clarifai.com/

Examples
## Not run:

# Before calling the function, set API secret and id via secret_id(c("client_id", "secret"))
# and get token via get_token()

get_langs()

## End(Not run)

get_token  Get Access Token

Description
Once you have set your Application ID and Secret via secret_id, get the access token.

Usage
get_token(...)

## Not run:
get_usage

Arguments
...

Additional arguments passed to clarifai_POST.

Value

a list with 4 things: access_token, token_type, expires_in, scope

References

https://developer.clarifai.com/

Examples

## Not run:

# Before calling the function, set API secret and id via secret_id(c("client_id", "secret"))

get_token()

## End(Not run)

---

get_usage

Get API usage for the current month and hour

Description

Get API usage for the current month and hour

Usage

get_usage(...)

Arguments

...

Additional arguments passed to clarifai_GET.

Value

named list with four items status_code, status_msg, results (user and application details — credits consumed, total credit)

References

https://developer.clarifai.com/
Examples

## Not run:

```r
# Before calling the function, set API secret and id via secret_id(c("client_id", "secret"))
# and get token via get_token()

gc_usage()

## End(Not run)
```

---

**secret_id**  
*Sets Application ID and Secret*

### Description

Set Client ID and Secret. Needed for interfacing with Clarifai. Run this before anything else.

### Usage

```r
secret_id(appdetails = NULL, force = FALSE)
```

#### Arguments

- `appdetails`  
  A vector of client_id, client_secret. Get these from [https://developer.clarifai.com/](https://developer.clarifai.com/). Set them before you use other functions.

- `force`  
  force reset client id and secret

#### Details

The function looks for `clarifaiId` and `clarifaiSecret` in the environment. If it doesn’t find them and if we don’t want to force change in them, it looks for arguments. And if no arguments are passed, it asks for user to input the values.

### References

- [https://developer.clarifai.com/](https://developer.clarifai.com/)

### Examples

## Not run:

```r
setapp(c("client_id", "client_secret"))

## End(Not run)
```
Tag Images on the Computer

Description
Tag Images on the Computer

Usage
tag_images(file_paths = NULL, model = NULL, language = NULL, meta = FALSE, simplify = TRUE, ...)

Arguments
- **file_paths**: a vector of paths to image file(s) for which you want tags
- **model**: For topic specific answers. Optional. Can be one of the following: general-v1.3, nsfw-v1.0, weddings-v1.0, travel-v1.0, food-items-v0.1. Default is NULL.
- **language**: Return tag in another language than english. 2 digit code. Optional. See list(get_langs) for the supported languages and their 2 digit codes.
- **meta**: Boolean that toggles whether or not you want the entire object.
- **simplify**: Boolean that toggles whether or not you want a simplified data frame with each label and associated probability in a separate row. Default is TRUE.
- **...**: Additional arguments passed to clarifai_POST.

The entire object returned by the API contains a lot of meta data. By default a simplified data.frame with img_url, associated labels, and probabilities is returned.

Value
If meta is TRUE, a named list of length 4 with following elements is returned: status_code OK or not status_msg Successful or not meta Named list of 1 containing another list named tag Sublist tag has three elements: timestamp, model and config results is a data.frame of length 6 and 1 row. Column names are: docid, status_code, status_msg, local_id and a data.frame named tag which has a data.frame result which contains two columns: labels and probabilities
If meta is FALSE and simplify is TRUE, a data.frame with three columns: file_paths, labels and probs returned
If meta is FALSE and simplify is FALSE, a data.frame with two columns carrying a vector of labels, vector of probs is returned for each image

References
https://developer.clarifai.com/

See Also
tag_image_urls
Examples

```r
## Not run:

# Before calling the function, set API secret and id via secret_id(c("client_id", "secret"))
# and get token via get_token()

tag_images(file_paths="path_to_image")

## End(Not run)
```

tag_image_urls  

**Tag Remote Images**

**Description**

Tag Remote Images

**Usage**

```r
tag_image_urls(img_urls = NULL, model = NULL, language = NULL,
meta = FALSE, simplify = TRUE, ...)
```

**Arguments**

- `img_urls`: A string or character vector of URLs of images for which you want tags
- `model`: For topic specific answers. Optional. Can be one of the following: general-v1.3, nsfw-v1.0, weddings-v1.0, travel-v1.0, food-items-v0.1 Default is NULL
- `language`: Return tag in another language than english. 2 digit code. Optional. See list(get_langs) for the supported languages and their 2 digit codes.
- `meta`: Boolean that toggles whether or not you want the entire object.
- `simplify`: Boolean that toggles whether or not you want a simplified data frame with each label and associated probability in a separate row. Default is TRUE.
- `...`: Additional arguments passed to `clarifai_post`.

The entire object returned by the API contains a lot of meta data. By default a simplified data.frame with `img_url`, associated labels, and probabilities is returned.

**Value**

If meta is TRUE, a named list of length 4 with following elements is returned: `status_code` OK or not `status_msg` Successful or not meta Named list of 1 containing another list named tag Sublist tag has three elements: `timestamp`, `model` and `config results` is a data.frame of length 6 and 1 row. Column names are: `docid`, `status_code`, `status_msg`, `local_id` and a data.frame named tag which has a data.frame result which contains two columns: labels and probabilities.

If meta is FALSE and simplify is TRUE, a data.frame with three columns: `img_urls`, `labels` and `probs` returned.
If `meta` is FALSE and `simplify` is FALSE, a data.frame with two columns carrying a vector of labels, vector of probs is returned for each image

**References**

https://developer.clarifai.com/

**See Also**

`tag_images`

**Examples**

```r
## Not run:

# Before calling the function, set API secret and id via secret_id(c("client_id", "secret"))
# and get token via get_token()

tag_image_urls(img_urls="url_of_image")
tag_image_urls("https://samples.clarifai.com/metro-north.jpg")
tag_image_urls("https://samples.clarifai.com/metro-north.jpg", language="es")
tag_image_urls("https://samples.clarifai.com/nsfw.jpg", model="nsfw-v1.0")

## End(Not run)
```
Index

* Topic **Client**
  secret_id, 10
* Topic **ID**
  secret_id, 10
* Topic **Secret**
  secret_id, 10
* Topic **Sets**
  secret_id, 10
* Topic **and**
  secret_id, 10

clarifai (clarifai-package), 2
clarifai-package, 2
clarifai_check_auth, 2
clarifai_check_results, 3
clarifai_check_token, 3
clarifai_GET, 3, 7–9
clarifai_POST, 4, 4, 5, 6, 9, 11, 12
curl_fetch_memory, 3, 4

feedback, 4
get_color, 5
get_color_urls, 6
get_info, 7
get_langs, 8
get_token, 2, 8
get_usage, 9

secret_id, 8, 10

tag_image_urls, 8, 11, 12
tag_images, 8, 11, 13