

Package: Rumblr (via r-universe)

September 27, 2026

Title Collecting and Analyzing 'Tumblr' Data

Version 0.2.0

Description An implementation of calls designed to collect 'Tumblr' data via its Application Program Interfaces (API), which can be found at the following URL:
<<https://www.tumblr.com/docs/en/api/v2>>.

License MIT + file LICENSE

URL <https://github.com/schochastics/Rumblr/>,
<https://schochastics.github.io/Rumblr/>

BugReports <https://github.com/schochastics/Rumblr/issues>

Encoding UTF-8

Roxygen list(markdown = TRUE)

Depends R (>= 2.10)

Imports dplyr, httr2, openssl, tibble

Suggests httptest2, httpuv, testthat (>= 3.0.0), withr

Config/testthat/edition 3

Config/roxygen2/version 8.0.0.9000

Config/pak/sysreqs libssl-dev

Repository <https://schochastics.r-universe.dev>

Date/Publication 2026-06-29 12:30:41 UTC

RemoteUrl <https://github.com/schochastics/Rumblr>

RemoteRef HEAD

RemoteSha 3334ada1d01d6d8d0898a319cd6bab1979cefdbe

Contents

get_blog_avatar	2
get_blog_blocks	3
get_blog_followed_by	4

get_blog_followers	4
get_blog_following	5
get_blog_info	6
get_blog_likes	7
get_blog_posts	8
get_dashboard	9
get_post	10
get_post_notes	11
get_posts_tag	12
get_user_following	13
get_user_info	14
get_user_likes	14
get_user_limits	15
rtumblr_auth	16
Index	17

get_blog_avatar	<i>Retrieve a Blog Avatar You can get a blog’s avatar in 9 different sizes</i>
-----------------	--

Description

Retrieve a Blog Avatar You can get a blog’s avatar in 9 different sizes

Usage

```
get_blog_avatar(blog, size = 64)
```

Arguments

- blog name of the blog
- size Integer. The size of the avatar (square, one value for both length and width). Must be one of the values: 16, 24, 30, 40, 48, 64, 96, 128, 512

Value

png of avatar

Examples

```
## Not run:  
avatar <- get_blog_avatar("schochastics")  
png::writePNG("avatar_schochastics.png")  
  
## End(Not run)
```

get_blog_blocks	<i>Retrieve Blogs Blocks</i>
-----------------	------------------------------

Description

Get the blogs that the requested blog is currently blocking. The requesting user must be an admin of the blog to retrieve this list. Note that this endpoint is rate limited to 60 requests per minute.

Usage

```
get_blog_blocks(  
  blog,  
  limit = 20,  
  offset = 0,  
  n = limit,  
  app_credentials = NULL  
)
```

Arguments

blog	name of the blog
limit	number of blocks to retrieve per request (1-20)
offset	block number to start at (default 0)
n	maximum number of blocks to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all.
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable

Value

tibble of blocked blogs

Examples

```
## Not run:  
# replace "your-blog-name" with your Tumblr username  
get_blog_blocks(blog = "your-blog-name")  
  
## End(Not run)
```

get_blog_followed_by	<i>Check If Followed By Blog This method can be used to check if one of your blogs is followed by another blog.</i>
----------------------	---

Description

Check If Followed By Blog This method can be used to check if one of your blogs is followed by another blog.

Usage

```
get_blog_followed_by(blog, query, app_credentials = NULL)
```

Arguments

blog	name of the blog
query	string. The name of the blog that may be following your blog
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable

Value

logical

Examples

```
## Not run:  
# replace "your-blog-name" with your Tumblr username  
get_blog_followed_by(blog = "your-blog-name", query = "blog-to-check")  
  
## End(Not run)
```

get_blog_followers	<i>Retrieve followers</i>
--------------------	---------------------------

Description

This method can be used to retrieve the publicly exposed list of blogs that follow a blog, in order from most recently-followed to first. **Only works with your own blog**

Usage

```
get_blog_followers(
  blog,
  limit = 20,
  offset = 0,
  n = limit,
  app_credentials = NULL,
  ...
)
```

Arguments

blog	name of the blog
limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Value

a tibble of blogs

Examples

```
## Not run:
# replace "your-blog-name" with your Tumblr username
get_blog_followers(blog = "your-blog-name")

## End(Not run)
```

get_blog_following	<i>Retrieve following</i>
--------------------	---------------------------

Description

This method can be used to retrieve the publicly exposed list of blogs that a blog follows, in order from most recently-followed to first. **Only works with your own account**

Usage

```
get_blog_following(
  blog,
  limit = 20,
  offset = 0,
  n = limit,
  app_credentials = NULL,
  ...
)
```

Arguments

blog	name of the blog
limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Value

a tibble of blogs

Examples

```
## Not run:
# replace "your-blog-name" with your Tumblr username
get_blog_following(blog = "your-blog-name")

## End(Not run)
```

get_blog_info

Retrieve Blog Info

Description

This method returns general information about the blog, such as the title, number of posts, and other high-level data.

Usage

```
get_blog_info(blog, api_key = NULL)
```

Arguments

blog	name of the blog
api_key	app consumer key. If NULL, attempts to load from the environment variable RTUMBLR_TOKEN

Value

tibble of information about blog

Examples

```
## Not run:
get_blog_info("schochastics")

## End(Not run)
```

get_blog_likes	<i>Retrieve Blog's Likes</i>
----------------	------------------------------

Description

This method can be used to retrieve the publicly exposed likes from a blog. **Seems to work only for your own blog**

Usage

```
get_blog_likes(
  blog,
  limit = 20,
  offset = 0,
  n = limit,
  after,
  before,
  api_key = NULL,
  ...
)
```

Arguments

blog	name of the blog
limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
after	integer. Retrieve posts liked after the specified timestamp

before	integer. Retrieve posts liked before the specified timestamp
api_key	app consumer key. If NULL, attempts to load from the environment variable RTUMBLR_TOKEN
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Details

You can only provide either before, after, or offset. If you provide more than one of these options together you will get an error. You can still use limit with any of those three options to limit your result set. When using the offset parameter the maximum limit on the offset is 1000. If you would like to get more results than that use either before or after.

Value

tibble of liked posts

Examples

```
## Not run:
# replace "your-blog-name" with your Tumblr username
get_blog_likes(blog = "your-blog-name")

## End(Not run)
```

get_blog_posts	<i>Retrieve Published Posts</i>
----------------	---------------------------------

Description

Retrieve Published Posts

Usage

```
get_blog_posts(blog, limit = 20, offset = 0, n = limit, api_key = NULL, ...)
```

Arguments

blog	name of the blog
limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
api_key	app consumer key. If NULL, attempts to load from the environment variable RTUMBLR_TOKEN
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Details

this function uses the new post format (npf: <https://www.tumblr.com/docs/npf>). Pagination is handled automatically: it pages through offset up to the API cap of 1000, then continues via the before timestamp, until n posts are collected or the blog is exhausted.

Value

a tibble of blog posts

Examples

```
## Not run:
# replace "blog-name" with a Tumblr username
get_blog_posts(blog = "blog-name")
# retrieve all posts of a blog
get_blog_posts(blog = "blog-name", n = Inf)

## End(Not run)
```

get_dashboard

Retrieve a User's Dashboard

Description

Returns the posts in the authenticating user's dashboard. Requires OAuth authentication (see [rtumblr_auth\(\)](#)).

Usage

```
get_dashboard(limit = 20, offset = 0, n = limit, app_credentials = NULL, ...)
```

Arguments

limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Value

a tibble of posts

Examples

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

get_post

Retrieve a Single Post

Description

Fetch a specific post by its id (new post format).

Usage

```
get_post(blog, post_id, api_key = NULL)
```

Arguments

blog	name of the blog
post_id	the id of the post to retrieve
api_key	app consumer key. If NULL, attempts to load from the environment variable RTUMBLR_TOKEN

Value

a tibble with the post

Examples

```
## Not run:  
get_post("blog-name", post_id = "1234567890")  
  
## End(Not run)
```

get_post_notes	<i>Retrieve Notes for a Post</i>
----------------	----------------------------------

Description

Get the notes (likes, reblogs, replies) for a specific post.

Usage

```
get_post_notes(  
  blog,  
  post_id,  
  mode = "all",  
  before_timestamp = NULL,  
  api_key = NULL  
)
```

Arguments

blog	name of the blog
post_id	the id of the post
mode	one of "all", "likes", "conversation", "rollup", or "reblogs_with_tags"
before_timestamp	fetch notes created before this timestamp (for pagination)
api_key	app consumer key. If NULL, attempts to load from the environment variable RTUMBLR_TOKEN

Value

a tibble of notes

Examples

```
## Not run:  
get_post_notes("blog-name", post_id = "1234567890")  
  
## End(Not run)
```

get_posts_tag

*Get Posts with Tag***Description**

Get Posts with Tag

Usage

```
get_posts_tag(tag, before, limit = 20, n = limit, api_key = NULL, ...)
```

Arguments

tag	tag to search for
before	the timestamp of when you'd like to see posts before
limit	number of results to return per request: 1-20
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to keep paging.
api_key	app consumer key. If NULL, attempts to load from the environment variable RTUMBLR_TOKEN
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Details

This function uses the legacy post format since it appears to not support the new post format. The /tagged endpoint does not support offset; pagination is done with the before timestamp, which is handled automatically when n is larger than limit.

Value

a list of tibbles of blog posts by format of posts

Examples

```
## Not run:
get_posts_tag(tag="meme")
# retrieve up to 100 posts across pages
get_posts_tag(tag="meme", n = 100)

## End(Not run)
```

get_user_following	<i>Retrieve the Blogs a User Is Following</i>
--------------------	---

Description

Returns the blogs followed by the authenticating user. Requires OAuth authentication (see `rtumblr_auth()`).

Usage

```
get_user_following(  
  limit = 20,  
  offset = 0,  
  n = limit,  
  app_credentials = NULL,  
  ...  
)
```

Arguments

limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Value

a tibble of blogs

Examples

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

get_user_info	<i>Retrieve a User's Information</i>
---------------	--------------------------------------

Description

Returns information about the authenticating user, including which blogs they own. Requires OAuth authentication (see [rtumblr_auth\(\)](#)).

Usage

```
get_user_info(app_credentials = NULL)
```

Arguments

app_credentials
a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable

Value

a tibble with the user's information; the user's blogs are stored in the blogs list column

Examples

```
## Not run:
get_user_info()

## End(Not run)
```

get_user_likes	<i>Retrieve a User's Likes</i>
----------------	--------------------------------

Description

Returns the posts liked by the authenticating user. Requires OAuth authentication (see [rtumblr_auth\(\)](#)).

Usage

```
get_user_likes(
  limit = 20,
  offset = 0,
  n = limit,
  after,
  before,
  app_credentials = NULL,
  ...
)
```

Arguments

limit	The number of results to return per request: 1-20
offset	post index to start at
n	maximum number of posts to retrieve across pages (default limit, i.e. a single page). Use Inf to retrieve all posts of the blog.
after	integer. Retrieve posts liked after the specified timestamp
before	integer. Retrieve posts liked before the specified timestamp
app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable
...	further parameters as described here: https://www.tumblr.com/docs/en/api/v2

Value

a tibble of liked posts

Examples

```
## Not run:
get_user_likes()

## End(Not run)
```

get_user_limits	<i>Retrieve a User's Limits</i>
-----------------	---------------------------------

Description

Returns the authenticating user's rate/usage limits. Requires OAuth authentication (see `rtumblr_auth()`).

Usage

```
get_user_limits(app_credentials = NULL)
```

Arguments

app_credentials	a named list containing the consumer key and consumer secret. If NULL, attempts to load from an env variable
-----------------	--

Value

a list with the user's limits as returned by the API

Examples

```
## Not run:
get_user_limits()

## End(Not run)
```

rtumblr_auth

Authenticate with Tumblr (OAuth 1.0a)

Description

Performs the interactive three-legged OAuth 1.0a flow: it opens a browser so you can authorize the app, then exchanges the verifier for a durable access token. The access token is cached on disk (under `tools::R_user_dir()`) so the browser only opens once; subsequent sessions reuse the cached token.

Usage

```
rtumblr_auth(app_credentials = NULL, cache = TRUE, force = FALSE)
```

Arguments

app_credentials	a named list with consumer_key and consumer_secret. If NULL, loaded from the RTUMBLR_TOKEN environment variable.
cache	logical, whether to read/write the access token from the on-disk cache.
force	logical, if TRUE re-run the flow even if a cached token exists.

Details

You normally do not need to call this directly: any function that hits an OAuth-protected endpoint will trigger it automatically when no cached token is found. As before, you only need to provide your app's consumer key and secret via the RTUMBLR_TOKEN environment variable ("consumer_key; consumer_secret").

Value

invisibly, a list with oauth_token and oauth_token_secret.

Examples

```
## Not run:
rtumblr_auth()

## End(Not run)
```

Index

`get_blog_avatar`, [2](#)
`get_blog_blocks`, [3](#)
`get_blog_followed_by`, [4](#)
`get_blog_followers`, [4](#)
`get_blog_following`, [5](#)
`get_blog_info`, [6](#)
`get_blog_likes`, [7](#)
`get_blog_posts`, [8](#)
`get_dashboard`, [9](#)
`get_post`, [10](#)
`get_post_notes`, [11](#)
`get_posts_tag`, [12](#)
`get_user_following`, [13](#)
`get_user_info`, [14](#)
`get_user_likes`, [14](#)
`get_user_limits`, [15](#)

`rtumblr_auth`, [16](#)
`rtumblr_auth()`, [9](#), [13–15](#)

`tools::R_user_dir()`, [16](#)