

# Package ‘RedditExtractor’

March 3, 2022

**Type** Package

**Title** Reddit Data Extraction Toolkit

**Version** 3.0.6

**Imports** RJSONIO, utils, rlang

**Depends** R (>= 4.1.0)

**Date** 2022-03-02

**Author** Ivan Rivera <ivan.s.rivera@gmail.com>

**Maintainer** Ivan Rivera <ivan.s.rivera@gmail.com>

**Description** A collection of tools for extracting structured data from <<https://www.reddit.com/>>.

**License** GPL-3

**RoxygenNote** 7.1.1

**Suggests** rmarkdown, knitr, mockery, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-03-03 12:00:02 UTC

## R topics documented:

|                              |   |
|------------------------------|---|
| find_subreddits . . . . .    | 2 |
| find_thread_urls . . . . .   | 2 |
| get_thread_content . . . . . | 3 |
| get_user_content . . . . .   | 4 |
| RedditExtractor . . . . .    | 4 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>6</b> |
|--------------|----------|

---

|                 |                                    |
|-----------------|------------------------------------|
| find_subreddits | <i>Find subreddits by keywords</i> |
|-----------------|------------------------------------|

---

**Description**

Search for subreddits and their attributes based on a keyword

**Usage**

```
find_subreddits(keywords)
```

**Arguments**

keywords      A string representing your search query

**Value**

A data frame with obtained redds

**Examples**

```
## Not run:  
find_subreddits("cats")  
  
## End(Not run)
```

---

|                  |                                |
|------------------|--------------------------------|
| find_thread_urls | <i>Find Reddit thread URLs</i> |
|------------------|--------------------------------|

---

**Description**

Find URLs to reddit threads of interest. There are 2 available search strategies: by keywords and by home page. Using a set of keywords Can help you narrow down your search to a topic of interest that crosses multiple subreddits whereas searching by home page can help you find, for example, top posts within a specific subreddit

**Usage**

```
find_thread_urls(  
  keywords = NA,  
  sort_by = "top",  
  subreddit = NA,  
  period = "month"  
)
```

**Arguments**

|           |  |
|-----------|--|
| keywords  | A optional string that you want to search for, e.g. "cute kittens". If NA, then either your front page will be searched or the front page of a specified subreddit   |
| sort_by   | A string representing how you want Reddit to sort the results. Note that this string is conditional on whether you are searching by keywords or not. If you are searching by keywords, then it must be one of: relevance, comments, new, hot, top; if you are not searching by keywords, then it must be one of: hot, new, top, rising |
| subreddit | (optional) A string representing the subreddit of interest   |
| period    | A string representing the period of interest (hour, day, week, month, year, all)   |

**Value**

a data frame with URLs to Reddit threads that are relevant to your input parameters

**Examples**

```
## Not run:
find_thread_urls(keywords="cute kittens", subreddit="cats", sort_by="new", period="month")
find_thread_urls(subreddit="cats", sort_by="rising", period="all")

## End(Not run)
```

---

get\_thread\_content      *Get thread contents of Reddit URLs*

---

**Description**

This function takes a collection of URLs and returns a list with 2 data frames: 1. a data frame containing meta data describing each thread 2. a data frame with comments found in all threads

**Usage**

```
get_thread_content(urls)
```

**Arguments**

|      |   |
|------|---|
| urls | A vector of strings pointing to a Reddit thread |
|------|---|

**Details**

The URLs are being retained in both tables which would allow you to join them if needed

**Value**

A list with 2 data frames "threads" and "comments"

get\_user\_content      *Find data relating to a vector of Reddit users*

---

### Description

Given a list of valid Reddit user names, obtain a list consisting of general information about each user, their comments and threads

### Usage

```
get_user_content(users)
```

### Arguments

users      A vector of strings representing valid Reddit user names

### Value

A nested list with user names containing another list that has "about" (list), "comments" (data frame) and "threads" (data frame)

### Examples

```
## Not run:  
get_user_content(c("memes", "nationalgeographic"))  
  
## End(Not run)
```

---

RedditExtractoR      *Reddit Data Extraction Toolkit*

---

### Description

Reddit is an online bulletin board and a social networking website where registered users can submit and discuss content. This package uses Reddit API to retrieve thread URLs, comments, subreddits and user information. For more information about the usage of this package, please see the following GitHub page: <https://github.com/ivan-rivera/RedditExtractoR>

### Details

Package: RedditExtractoR  
Type: Package  
Version: 3.0.0  
Date: 2015-06-14  
License: GPL-3

The package contains a collection of functions for extracting threads of interest and their corresponding comments, as well as functions for analysing the structure of these threads.

**Author(s)**

Ivan Rivera

Maintainer: Ivan Rivera <ivan.s.rivera@gmail.com>

**See Also**

<https://www.reddit.com>

# Index

## \* **reddit**

RedditExtractor, 4

find\_subreddits, 2

find\_thread\_urls, 2

get\_thread\_content, 3

get\_user\_content, 4

RedditExtractor, 4