

Package ‘rscopus’

October 12, 2017

Type Package

Title Scopus Database 'API' Interface

Version 0.5.3

Date 2017-10-11

Maintainer John Muschelli <muschelli2@gmail.com>

Description Uses Elsevier 'Scopus' 'API'
<https://dev.elsevier.com/sc_apis.html> to download
information about authors and their citations.

License GPL-2

Depends R (>= 3.2.0)

Imports httr, utils, stats, plyr, tidyr

LazyData TRUE

Suggests xml2, rvest, graphics, testthat, jpeg

URL https://dev.elsevier.com/sc_apis.html,
<https://github.com/muschelli2/rscopus>

BugReports <https://github.com/muschelli2/rscopus/issues>

RoxygenNote 6.0.1.9000

Encoding UTF-8

NeedsCompilation no

Author John Muschelli [aut, cre]

Repository CRAN

Date/Publication 2017-10-11 22:57:25 UTC

R topics documented:

abstract_retrieval	2
affiliation_retrieval	3
affil_df	4
affil_list_to_df	5

affil_search	5
all_possible_affils	6
article_retrieval	6
author_df	7
author_retrieval_id	8
author_search	9
collapse_affil	10
embase_retrieval	11
entitlement_retrieval	12
entries_to_affil_list	13
entries_to_citation_df	13
entries_to_df	14
entry_to_affil	14
generic_elsevier_api	15
get_api_key	16
get_author_info	17
get_complete_author_info	17
nonnull	18
object_retrieval	18
process_author_name	19
recommendation_retrieval	20
replace_non_ascii	21
set_api_key	21
Index	22

abstract_retrieval	<i>SCOPUS Abstract Retrieval</i>
--------------------	----------------------------------

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an abstract from the Elsevier Abstract Retrieval API

Usage

```
abstract_retrieval(id, identifier = c("scopus_id", "eid", "doi", "pii",
  "pubmed_id", "pui", "group_id"), http_end = NULL, ...)
```

Arguments

id	Identifier for abstract
identifier	Type of identifier to use
http_end	any additional end to http statement. See generic_elsevier_api
...	Arguments to be passed to generic_elsevier_api

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)
if (!is.null(api_key)){
  x = abstract_retrieval("S1053811915002700", identifier = "pii",
    verbose = FALSE)
}
```

affiliation_retrieval *SCOPUS Affiliation Retrieval*

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an affiliation from the Elsevier Affiliation Retrieval API

Usage

```
affiliation_retrieval(id, identifier = c("affiliation_id", "eid"),
  http_end = NULL, ...)
```

Arguments

<code>id</code>	Identifier for affiliation
<code>identifier</code>	Type of identifier to use
<code>http_end</code>	any additional end to http statement. See generic_elsevier_api
<code>...</code>	Arguments to be passed to generic_elsevier_api

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)
if (!is.null(api_key)){
  x = affiliation_retrieval("60006183", identifier = "affiliation_id",
    verbose = FALSE)
}
```

affil_df

Search Affiliation Content on SCOPUS

Description

Searches SCOPUS to get information about documents on an affiliation

Usage

```
affil_df(affil_id, verbose = TRUE, ...)
```

```
affil_data(affil_id, verbose = TRUE, ...)
```

Arguments

affil_id	Affiliation ID number.
verbose	Print diagnostic messages
...	Arguments to be passed to author_search

Value

List of entries from SCOPUS

Note

The `affil_data` command will return the list of all entries as well as the `data.frame`

See Also

[get_author_info](#)

affil_list_to_df	<i>List of SCOPUS Entries to List of Affiliations Data Frames</i>
------------------	---

Description

Take a SCOPUS entry and transform it to a data frame of affiliations

Usage

```
affil_list_to_df(affils)
```

Arguments

affils List of affiliations, from [entries_to_affil_list](#)

Value

A data.frame of affiliation information. A column named "index" denotes the element of the object affils that the row corresponds to

affil_search	<i>Search Author Content on SCOPUS</i>
--------------	--

Description

Searches SCOPUS to get information about documents on an author.

Usage

```
affil_search(affil_id, searcher = "AF-ID", ...)
```

Arguments

affil_id Affiliation ID number
searcher Identifier for Affiliation ID. Do not change unless you know exactly what the API calls for.
... Arguments to be passed to [GET](#)

Value

List of entries from SCOPUS

See Also

[get_author_info](#)

`all_possible_affils` *Find all affiliations*

Description

Take a SCOPUS entry and transform it to a data frame of all affiliations listed in there

Usage

```
all_possible_affils(entries)
```

Arguments

`entries` list of entries from SCOPUS, usually from [author_search](#) result

Value

A data.frame of affiliations

`article_retrieval` *ScienceDirect Article Retrieval*

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an article from the Elsevier Article Retrieval API

Usage

```
article_retrieval(id, identifier = c("scopus_id", "eid", "doi", "pii",
  "pubmed_id"), http_end = NULL, ...)
```

Arguments

`id` Identifier for article
`identifier` Type of identifier to use
`http_end` any additional end to http statement. See [generic_elsevier_api](#)
`...` Arguments to be passed to [generic_elsevier_api](#)

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```

api_key = get_api_key(NULL, error = FALSE)
if (!is.null(api_key)){
  x = article_retrieval("S1053811915002700", identifier = "pii",
    verbose = FALSE)
} else {
  x = article_retrieval("S1053811915002700",
    identifier = "pii",
    api_key_error = FALSE)
}

```

author_df

*Search Author Content on SCOPUS***Description**

Searches SCOPUS to get information about documents on an author.

Usage

```
author_df(au_id, last_name, first_name, api_key = NULL, verbose = TRUE,
  all_author_info = FALSE, ...)
```

```
author_data(au_id, last_name, first_name, api_key = NULL, verbose = TRUE,
  all_author_info = FALSE, ...)
```

Arguments

au_id	Author ID number. Overrides any first/last name argument
last_name	last name of author
first_name	first name of author
api_key	Elsevier API key
verbose	Print diagnostic messages
all_author_info	Should all author info be recorded instead of that just to the author given
...	Arguments to be passed to author_search

Value

List of entries from SCOPUS

Note

The author_data command will return the list of all entries as well as the data.frame.

See Also[get_author_info](#)**Examples**

```
## Not run:
author_df(last_name = "Muschelli", first_name = "John", verbose = FALSE)

## End(Not run)
```

author_retrieval_id *SCOPUS Author Retrieval*

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an author from the Elsevier Author Retrieval API

Usage

```
author_retrieval_id(id, identifier = c("author_id", "eid"), http_end = NULL,
  ...)

multi_author_retrieval(id, identifier = c("author_id", "eid"),
  http_end = NULL, ...)

author_retrieval(au_id, last_name, first_name, view = c("LIGHT", "STANDARD",
  "ENHANCED", "METRICS", "ENTITLED"), self_cite = c("include", "exclude"),
  http_end = NULL, verbose = TRUE, api_key = NULL, ...)
```

Arguments

id	Identifier for author
identifier	Type of identifier to use
http_end	any additional end to http statement. See generic_elsevier_api
...	Arguments to be passed to generic_elsevier_api
au_id	Author ID number. Overrides any first/last name argument
last_name	last name of author
first_name	first name of author
view	Which view to see. See https://api.elsevier.com/documentation/AuthorRetrievalAPI.wadl
self_cite	Should self-citations be included?
verbose	Print diagnostic messages
api_key	Elsevier API key

Value

List of elements, similar to [generic_elsevier_api](#)

Note

See <https://api.elsevier.com/documentation/AuthorRetrievalAPI.wadl> for documentation

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)
if (!is.null(api_key)){
  x = author_retrieval(au_id = "40462056100",
    verbose = FALSE)
  x = author_retrieval_id("40462056100", identifier = "author_id",
    verbose = FALSE)
} else {
  x = author_retrieval_id(
    "40462056100",
    identifier = "author_id",
    api_key_error = FALSE, verbose = FALSE)
  x = author_retrieval(
    au_id = "40462056100",
    api_key_error = FALSE, verbose = FALSE)
}
```

author_search

Search Author Content on SCOPUS

Description

Searches SCOPUS to get information about documents on an author.

Usage

```
author_search(au_id, api_key = NULL,
  http = "http://api.elsevier.com/content/search/scopus", count = 25,
  verbose = TRUE, facets = "subjarea(sort=fd)", searcher = "AU-ID",
  max_count = Inf, ...)
```

Arguments

au_id	Author ID number
api_key	API Key for Elsevier
http	Address for scopus api
count	number of records to retrieve (below 25, see http://dev.elsevier.com/api_key_settings.html)
verbose	Print diagnostic messages
facets	Facets sent in query. See http://dev.elsevier.com/api_docs.html
searcher	Identifier for author ID. Do not change unless you know exactly what the API calls for.
max_count	Maximum count of records to be returned.
...	Arguments to be passed to the query list for GET

Value

List of entries from SCOPUS

See Also

[get_author_info](#)

collapse_affil	<i>Collapse Affiliations to one row</i>
----------------	---

Description

Take an individual SCOPUS entry and transform it to a data frame of affiliations

Usage

```
collapse_affil(affils, collapse = ";")
```

Arguments

affils	Data frame of affiliations to be collapsed usually from author_search
collapse	What should values be collapsed using as a separator

Value

A data.frame of affiliation information

embase_retrieval	<i>SCOPUS Embase Retrieval</i>
------------------	--------------------------------

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an Embase Entry from the Elsevier Embase Retrieval API

Usage

```
embase_retrieval(id, identifier = c("lui", "pii", "doi", "embase",  
  "pubmed_id", "medline"), http_end = NULL, ...)
```

Arguments

id	Identifier for abstract
identifier	Type of identifier to use
http_end	any additional end to http statement. See generic_elsevier_api
...	Arguments to be passed to generic_elsevier_api

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)  
if (!is.null(api_key)){  
  x = embase_retrieval("S1053811915002700", identifier = "pii",  
    verbose = FALSE)  
}
```

entitlement_retrieval *ScienceDirect Text Entitlement Retrieval*

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an entitlement from the Elsevier Text Entitlement API

Usage

```
entitlement_retrieval(id, identifier = c("scopus_id", "eid", "doi", "pii",  
  "pubmed_id"), http_end = NULL, ...)
```

Arguments

<code>id</code>	Identifier for entitlement
<code>identifier</code>	Type of identifier to use
<code>http_end</code>	any additional end to http statement. See generic_elsevier_api
<code>...</code>	Arguments to be passed to generic_elsevier_api

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)  
if (!is.null(api_key)){  
  x = entitlement_retrieval("S1053811915002700", identifier = "pii",  
    verbose = FALSE)  
}
```

entries_to_affil_list *List of SCOPUS Entries to List of Affiliations Data Frames*

Description

Take a SCOPUS entry and transform it to a data frame of affiliations

Usage

```
entries_to_affil_list(entries)
```

Arguments

entries list of entries from SCOPUS, usually from [author_search](#) result

Value

A data.frame of affiliation information

entries_to_citation_df
List of SCOPUS Entries to Data Frame of Citations

Description

Take a SCOPUS entry list and transform it to a data frame of citation and article information

Usage

```
entries_to_citation_df(entries)
```

Arguments

entries list of entries from SCOPUS, usually from [author_search](#) result

Value

A data.frame of citation information

entries_to_df *Convert Entries into a data.frame*

Description

Converts a list of entries into a data.frame of records

Usage

```
entries_to_df(entries, au_id = NULL, verbose = TRUE)
```

Arguments

entries	Author ID number. Overrides any first/last name argument
au_id	Author ID number. Overrides any first/last name argument
verbose	Print diagnostic messages
...	Arguments to be passed to author_search

Value

Data frame of records

entry_to_affil *SCOPUS Entry to Affiliation*

Description

Take an individual SCOPUS entry and transform it to a data frame of affiliations

Usage

```
entry_to_affil(x, all_affils)
```

Arguments

x	individual entry from SCOPUS, usually from author_search
all_affils	Affiliation data.frame from all_possible_affils

Value

A data.frame of affiliation information

generic_elsevier_api *Generic Elsevier Search*

Description

Runs GET on generic Elsevier Search

Usage

```
generic_elsevier_api(query = NULL, type = c("search", "article",
  "entitlement", "recommendation", "object", "fragment", "abstract",
  "affiliation", "embase", "author", "serial", "nonserial", "subject",
  "holdings", "citation-count", "citations"), search_type = c("affiliation",
  "author", "scopus", "scidir", "scidir-object"), api_key = NULL,
  headers = NULL, content_type = c("content", "feedback"),
  root_http = "http://api.elsevier.com", http_end = NULL, verbose = TRUE,
  api_key_error = TRUE, ...)
```

Arguments

query	Query to run
type	Type of search. See http://dev.elsevier.com/api_docs.html
search_type	Type of search if type = "search". See http://dev.elsevier.com/api_docs.html
api_key	Elsevier API key
headers	Headers passed to add_headers , passed to GET
content_type	Is the data content or feedback?
root_http	address to use for query
http_end	string to add to end of http specification (done using paste0)
verbose	Print messages from specification
api_key_error	Should there be an error if no API key?
...	Options passed to query for GET

Value

List of elements, content and the GET request

Examples

```
## Not run:
query_string = "affil(hopkins)"
# Use affiliation query
s = generic_elsevier_api(query = query_string,
  type = "search", search_type = "affiliation",
  api_key = api_key,
```

```

        verbose = FALSE)

# Use author query
s = generic_elsevier_api(query = query_string,
  type = "search", search_type = "author",
  api_key = api_key,
  verbose = FALSE)

# Query abstract by pii
s = generic_elsevier_api(query = "",
  type = "abstract", http_end = "pii/S1053811915002700",
  api_key = api_key,
  verbose = FALSE)

## End(Not run)

```

get_api_key

Find API Key for Elsevier

Description

Determines if `option(elsevier_api_key)` or `option(elsevier_api_key_filename)` is set. If not, it stops and returns an error. If so, returns the value.

Usage

```
get_api_key(api_key = NULL, error = TRUE)
```

```
have_api_key(api_key = NULL)
```

Arguments

api_key	Elsevier API key
error	Should the function error if api_key = NULL?

Value

API key

Note

You can either set the API key using `option(elsevier_api_key)` or have it accessible by `api_key = Sys.getenv('Elsevier')`

Examples

```
res = get_api_key(error = FALSE)
```

get_author_info	<i>Get Relevant Author Information and ID from Scopus in Data Frame</i>
-----------------	---

Description

Uses SCOPUS author search to identify author identification information in a workable format

Usage

```
get_author_info(...)
```

Arguments

... Arguments passed to [get_complete_author_info](#)

Value

Data.frame of information

See Also

[get_complete_author_info](#)

get_complete_author_info	<i>Get Complete Author Information and ID from Scopus</i>
--------------------------	---

Description

Uses SCOPUS author search to identify author identification information

Usage

```
get_complete_author_info(last_name, first_name = NULL, api_key = NULL,
  http = "http://api.elsevier.com/content/search/author", query = NULL,
  verbose = TRUE, ...)
```

Arguments

last_name	last name of author
first_name	first name of author
api_key	Elsevier API key
http	Author API http
query	Additional query info, added using +AND+ to original query
verbose	Print messages from specification
...	options to pass to GET

Value

List of information

nonnull	<i>Remove NULL</i>
---------	--------------------

Description

Removes NULL values from a vector from a list

Usage

```
nonnull(x, replace = NA)
```

Arguments

x	Vector from a list
replace	Value to replace NULL with

Value

Vector

object_retrieval	<i>ScienceDirect Object Retrieval</i>
------------------	---------------------------------------

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of an object from the Elsevier Object Retrieval API

Usage

```
object_retrieval(id, identifier = c("scopus_id", "eid", "doi", "pii",
  "pubmed_id"), ref = NULL, http_end = NULL, ...)
```

Arguments

id	Identifier for object
identifier	Type of identifier to use
ref	document reference
http_end	any additional end to http statement. See generic_elsevier_api
...	Arguments to be passed to generic_elsevier_api

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)
if (!is.null(api_key)){
  x = object_retrieval("S1053811915002700", identifier = "pii",
    verbose = FALSE)
  if (requireNamespace("xml2") & requireNamespace("httr") &
    requireNamespace("rvest"))
  ){
    refs = httr::content(x$get_statement, "text")
    refs = xml2::read_xml(refs)
    refs = rvest::xml_nodes(refs, "choice")
    texts = xml2::xml_text(refs)
    types = xml2::xml_attr(refs, "type")
    refs = xml2::xml_attr(refs, "ref")
    df = data.frame(ref = refs, type = types, text = texts,
      stringsAsFactors = FALSE)
    df = df[ grepl("image/jpeg", df$text),,drop = FALSE ]
    df = df[ df$type %in% "IMAGE-HIGH-RES",,drop = FALSE ]
    #r = httr::GET(url = df$text[1],
    #      query = list("apiKey" = api_key))
    query = paste0("apiKey=", api_key)
    r = httr::GET(paste0(df$text[1], "&", query))
    img = httr::content(r)
    dims = dim(img)[1:2]
    mdim = max(dims)
    graphics::plot(c(0, ncol(img)), c(0, nrow(img)), type='n')
    graphics::rasterImage(img, 1, 1, ncol(img), nrow(img))
  }
  obj = object_retrieval('S1053811915002700', "pii", verbose = FALSE)
}
```

process_author_name *Process Author Name*

Description

Process author ID and names for generic use

Usage

```
process_author_name(au_id, last_name, first_name, api_key = NULL,
  verbose = TRUE)
```

Arguments

au_id	Author ID number. Overrides any first/last name argument
last_name	last name of author
first_name	first name of author
api_key	Elsevier API key
verbose	Print diagnostic messages

Value

List of first/last name and author ID

Note

This function is really to avoid duplication

recommendation_retrieval

ScienceDirect Article Recommendation Retrieval

Description

This function wraps [generic_elsevier_api](#) to give a retrieval of a recommendation from the Elsevier Article Recommendation API

Usage

```
recommendation_retrieval(id, identifier = c("eid", "pii"), http_end = NULL,  
...)
```

Arguments

id	Identifier for recommendation
identifier	Type of identifier to use
http_end	any additional end to http statement. See generic_elsevier_api
...	Arguments to be passed to generic_elsevier_api

Value

List of elements, similar to [generic_elsevier_api](#)

See Also

[generic_elsevier_api](#)

Examples

```
api_key = get_api_key(NULL, error = FALSE)
if (!is.null(api_key)){
  x = recommendation_retrieval("S1053811915002700", identifier = "pii",
  verbose = FALSE)
}
```

replace_non_ascii *Replace non-ASCII characters*

Description

Replaces non-ASCII characters from a last or first name

Usage

```
replace_non_ascii(string)
```

Arguments

string Character vector of values to be replaced

Value

Character vector

See Also

Taken from <http://stackoverflow.com/questions/20495598/replace-accented-characters-in-r-with-non-acc>

set_api_key *Set API Key for Elsevier*

Description

Sets Elsevier API key using if option(elsevier_api_key)

Usage

```
set_api_key(api_key)
```

Arguments

api_key Elsevier API key

Index

abstract_retrieval, [2](#)
add_headers, [15](#)
affil_data (affil_df), [4](#)
affil_df, [4](#)
affil_list_to_df, [5](#)
affil_search, [5](#)
affiliation_retrieval, [3](#)
all_possible_affils, [6](#), [14](#)
article_retrieval, [6](#)
author_data (author_df), [7](#)
author_df, [7](#)
author_retrieval (author_retrieval_id),
[8](#)
author_retrieval_id, [8](#)
author_search, [4](#), [6](#), [7](#), [9](#), [10](#), [13](#), [14](#)

collapse_affil, [10](#)

embase_retrieval, [11](#)
entitlement_retrieval, [12](#)
entries_to_affil_list, [5](#), [13](#)
entries_to_citation_df, [13](#)
entries_to_df, [14](#)
entry_to_affil, [14](#)

generic_elsevier_api, [2](#), [3](#), [6](#), [8](#), [9](#), [11](#), [12](#),
[15](#), [18–20](#)
GET, [5](#), [10](#), [15](#), [17](#)
get_api_key, [16](#)
get_author_info, [4](#), [5](#), [8](#), [10](#), [17](#)
get_complete_author_info, [17](#), [17](#)

have_api_key (get_api_key), [16](#)

multi_author_retrieval
 (author_retrieval_id), [8](#)

nonnull, [18](#)

object_retrieval, [18](#)

process_author_name, [19](#)

recommendation_retrieval, [20](#)
replace_non_ascii, [21](#)

set_api_key, [21](#)