

Package ‘rscopus’

May 24, 2018

Type Package

Title Scopus Database 'API' Interface

Version 0.5.11

Date 2018-05-24

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, dplyr

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

Encoding UTF-8

NeedsCompilation no

Author John Muschelli [aut, cre]

Repository CRAN

Date/Publication 2018-05-24 18:42:41 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	10
author_search_by_affil	11
collapse_affil	12
complete_multi_author_info	12
embase_retrieval	13
entitlement_retrieval	14
entries_to_affil_list	15
entries_to_citation_df	15
entries_to_df	16
entry_to_affil	16
generic_elsevier_api	17
gen_entries_to_df	18
get_api_key	18
get_author_info	19
get_complete_author_info	20
get_links	20
multi_author_info	22
nonnull	22
object_retrieval	23
process_affiliation_name	24
process_author_name	25
recommendation_retrieval	25
replace_non_ascii	26
scopus_search	27
set_api_key	28
subject_areas	28
Index	29

abstract_retrieval *SCOPUS Abstract Retrieval*

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

id	Identifier for affiliation
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 = 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 = NULL, affil_name = NULL, api_key = NULL,
  verbose = TRUE, facets = NULL, sort = "document-count", ...)
```

```
affil_data(affil_id = NULL, affil_name = NULL, api_key = NULL,
  verbose = TRUE, facets = NULL, sort = "document-count", ...)
```

Arguments

affil_id	Affiliation ID number.
affil_name	affiliation name. Disregarded if affil_id is specified
api_key	Elsevier API key
verbose	Print diagnostic messages
facets	Facets sent in query. See https://dev.elsevier.com/api_docs.html
sort	sorting sent to query
...	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 *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

```
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 *Search Author Content on SCOPUS*

Description

Searches SCOPUS to get information about documents on an author.

Usage

```
affil_search(affil_id, searcher = "AF-ID",
             http = "https://api.elsevier.com/content/search/affiliation",
             facets = "affilcountry(sort=document-count)", count = 200, ...)
```

Arguments

affil_id Affiliation ID number

searcher Identifier for Affiliation ID. Do not change unless you know exactly what the API calls for.

http Address for scopus api

facets Facets sent in query. See https://dev.elsevier.com/api_docs.html

count number of records to retrieve (below 200, see https://dev.elsevier.com/api_key_settings.html)

... Arguments to be passed to [author_search](#)

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. Note, `author_list` returns a list of the entries from `author_search`, but allows you to put in a name.

Usage

```
author_df(au_id, last_name, first_name, api_key = NULL, verbose = TRUE,
  all_author_info = FALSE,
  http = "https://api.elsevier.com/content/search/scopus",
  view = "COMPLETE", count = 25, general = TRUE, ...)

author_df_orig(..., general = FALSE)

author_list(au_id, last_name, first_name, api_key = NULL, verbose = TRUE,
  http = "https://api.elsevier.com/content/search/scopus",
  view = "COMPLETE", count = 25, ...)

author_data(..., verbose = TRUE, all_author_info = FALSE, general = TRUE)
```

Arguments

<code>au_id</code>	Author ID number. Overrides any first/last name argument
<code>last_name</code>	last name of author
<code>first_name</code>	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
http	Address for scopus api
view	type of view to give, see https://api.elsevier.com/documentation/ScopusSearchAPI.wadl
count	number of records to retrieve (below 25, see https://dev.elsevier.com/api_key_settings.html)
general	Should gen_entries_to_df instead of the way before version 0.5.10.9001
...	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 = "https://api.elsevier.com/content/search/author", count = 200,
  start = 0, verbose = TRUE, facets = "subjarea(sort=fd)",
  searcher = "AU-ID", max_count = Inf, view = c("STANDARD", "COMPLETE"),
  add_query = NULL, ...)

```

Arguments

au_id	Author ID number
api_key	API Key for Elsevier
http	Address for scopus api
count	number of records to retrieve (below 200, see https://dev.elsevier.com/api_key_settings.html)
start	where should the records start gathering
verbose	Print diagnostic messages
facets	Facets sent in query. See https://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.
view	type of view to give, see https://api.elsevier.com/documentation/AuthorSearchAPI.wadl
add_query	Things to add to the query parameter for the request
...	Arguments to be passed to the query list for GET

Value

List of entries from SCOPUS

See Also

[get_author_info](#)

Examples

```
## Not run:
author_search(au_id = "Smith", searcher = "affil(princeton) and authlast")
berk = author_search(au_id = "berkeley", searcher = "affil", count =100)

## End(Not run)
```

author_search_by_affil

Search Authors by Affiliation on SCOPUS

Description

Searches SCOPUS to get information about authors with a certain affiliation

Usage

```
author_search_by_affil(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](#)

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

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

Description

Get Complete Author Information and ID from Scopus

Usage

```
complete_multi_author_info(au_id = NULL, api_key = NULL, verbose = TRUE,  
  ...)
```

Arguments

au_id	vector of Author IDs
api_key	Elsevier API key
verbose	Print messages from specification
...	options to pass to generic_elsevier_api

Value

List of 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

id	Identifier for entitlement
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 = 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)
```

```
entries_to_df2(entries, verbose = TRUE)
```

Arguments

entries	Entries from the output of a command
au_id	Author ID number. Overrides any first/last name argument
verbose	Print diagnostic messages

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 = "https://api.elsevier.com", http_end = NULL, verbose = TRUE,
  api_key_error = TRUE, ...)
```

Arguments

query	Query to run, not overall query, but ‘queryParam‘ query
type	Type of search. See https://dev.elsevier.com/api_docs.html
search_type	Type of search if type = "search". See https://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 queryParam 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)

```

gen_entries_to_df *Generally Convert Entries into a list of data.frames*

Description

Generally Convert Entries into a list of data.frames

Usage

```
gen_entries_to_df(entries)
```

Arguments

entries Entries from the output of a command

Value

List of data.frames from entries

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 *Get Relevant Author Information and ID from Scopus in Data Frame*

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](#)

Examples

```
## Not run:  
get_author_info(au_id = "40462056100")  
  
## End(Not run)
```

get_complete_author_info

Get Complete Author Information and ID from Scopus

Description

Uses SCOPUS author search to identify author identification information

Usage

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

Arguments

last_name	last name of author
first_name	first name of author
affil_id	ID of affiliation (optional)
api_key	Elsevier API key
http	Author API http
query	Additional query info, added using +AND+ to original query
verbose	Print messages from specification
au_id	Author ID number, will override first/last combination if specified
...	options to pass to GET

Value

List of information

get_links

Get Links for next/first/last query

Description

Get Links for next/first/last query

Usage

```
get_links(result)

get_url(result, type = c("self", "first", "prev", "next", "last"))

next_url(result)

last_url(result)

prev_url(result)

self_url(result)

first_url(result)

get_link_type(result, ..., type = c("self", "first", "prev", "next", "last"))

get_first(result, ...)

get_last(result, ...)

get_prev(result, ...)

get_next(result, ...)

get_self(result, ...)
```

Arguments

result	Object (list) with an element named content, usually from generic_elsevier_api
type	The type of link requested
...	Options passed to GET

Value

A data.frame or a vector of characters

Examples

```
## Not run:
result <- generic_elsevier_api(
  query = "ISSN(0004-3702) AND YEAR(2001)",
  search_type = "scopus")
next_result = get_next(result)

## End(Not run)
```

multi_author_info	<i>Get Relevant Authors Information from IDs from Scopus</i>
-------------------	--

Description

Get Relevant Authors Information from IDs from Scopus

Usage

```
multi_author_info(...)
```

Arguments

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

Value

Data.frame of information

See Also

[get_complete_author_info](#)

Examples

```
## Not run:  
multi_author_info(au_id = "22968535800", "40462056100")  
  
## End(Not run)
```

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$get_statement)
    refs = refs$choices
    df = t(sapply(refs$choice, c))
    cn = colnames(df)
    cn[ grep("ref", cn)] = "ref"
    cn[ grep("type", cn)] = "type"
    cn[cn == "$"] = "text"
    colnames(df) = cn
    df = as.data.frame(df, 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_affiliation_name

Process Affiliation Name

Description

Process Affiliation ID and names for generic use

Usage

```
process_affiliation_name(affil_id = NULL, affil_name = NULL,
                        api_key = NULL, verbose = TRUE)
```

Arguments

affil_id	Affiliation ID number.
affil_name	affiliation name
api_key	Elsevier API key
verbose	Print diagnostic messages

Value

List of affiliation name and affiliation ID

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	<i>Replace non-ASCII characters</i>
-------------------	-------------------------------------

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 <https://stackoverflow.com/questions/20495598/replace-accented-characters-in-r-with-non-ac>

scopus_search	<i>SCOPUS Search</i>
---------------	----------------------

Description

This function wraps [generic_elsevier_api](#) to give a scopus search from the Elsevier Scopus Search API

Searches SCOPUS to get information about documents on an author.

Usage

```
scopus_search(query, api_key = NULL, count = 200, view = c("STANDARD",
  "COMPLETE"), start = 0, verbose = TRUE, max_count = 20000,
  http = "https://api.elsevier.com/content/search/scopus", ...)
```

Arguments

query	Query string to search on SCOPUS
api_key	API Key for Elsevier
count	number of records to retrieve (below 200 for STANDARD, below 25 for COMPLETE views, see https://dev.elsevier.com/api_key_settings.html)
view	type of view to give, see https://api.elsevier.com/documentation/ScopusSearchAPI.wadl
start	where should the records start gathering
verbose	Print diagnostic messages
max_count	Maximum count of records to be returned.
http	Address for scopus API
...	Arguments to be passed to the query list for GET

Value

List of entries from SCOPUS

Examples

```
## Not run:
res = scopus_search(query = "all(gene)", max_count = 200)
df = gen_entries_to_df(res$entries)

## End(Not run)
```

set_api_key	<i>Set API Key for Elsevier</i>
-------------	---------------------------------

Description

Sets Elsevier API key using `if option(elsevier_api_key)`

Usage

```
set_api_key(api_key)
```

Arguments

api_key	Elsevier API key
---------	------------------

subject_areas	<i>Subject Areas</i>
---------------	----------------------

Description

Subject Areas

Usage

```
subject_areas()
```

Value

Character vector of subject areas

Index

abstract_retrieval, 2
add_headers, 17
affil_data (affil_df), 4
affil_df, 4
affil_list_to_df, 5
affil_search, 5
affiliation_retrieval, 3
all_possible_affils, 6, 16
article_retrieval, 6
author_data (author_df), 7
author_df, 7
author_df_orig (author_df), 7
author_list (author_df), 7
author_retrieval (author_retrieval_id),
8
author_retrieval_id, 8
author_search, 4–6, 8, 10, 12, 15, 16
author_search_by_affil, 11

collapse_affil, 12
complete_multi_author_info, 12

embase_retrieval, 13
entitlement_retrieval, 14
entries_to_affil_list, 5, 15
entries_to_citation_df, 15
entries_to_df, 16
entries_to_df2 (entries_to_df), 16
entry_to_affil, 16

first_url (get_links), 20

gen_entries_to_df, 8, 18
generic_elsevier_api, 2–4, 6–9, 12–14, 17,
23, 25–27
GET, 10, 11, 17, 20, 21, 27
get_api_key, 18
get_author_info, 4, 6, 8, 11, 19
get_complete_author_info, 19, 20, 22
get_first (get_links), 20
get_last (get_links), 20
get_link_type (get_links), 20
get_links, 20
get_next (get_links), 20
get_prev (get_links), 20
get_self (get_links), 20
get_url (get_links), 20

have_api_key (get_api_key), 18

last_url (get_links), 20

multi_author_info, 22
multi_author_retrieval
(author_retrieval_id), 8

next_url (get_links), 20
nonnull, 22

object_retrieval, 23

prev_url (get_links), 20
process_affiliation_name, 24
process_author_name, 25

recommendation_retrieval, 25
replace_non_ascii, 26

scopus_search, 27
self_url (get_links), 20
set_api_key, 28
subject_areas, 28