

Package ‘dimensionsR’

August 28, 2020

Title Gathering Bibliographic Records from 'Digital Science Dimensions' Using 'DSL' API

Version 0.0.2

Description A set of tools to extract bibliographic content from 'Digital Science Dimensions' using 'DSL' API <<https://www.dimensions.ai/dimensions-apis/>>.

License GPL-3

URL <https://github.com/massimoaria/dimensionsR>

BugReports <https://github.com/massimoaria/dimensionsR/issues>

Encoding UTF-8

LazyData true

Imports httr, jsonlite

Suggests bibliometrix, knitr, rmarkdown

RoxygenNote 7.1.0

VignetteBuilder knitr

NeedsCompilation no

Author Massimo Aria [aut, cre] (<<https://orcid.org/0000-0002-8517-9411>>)

Maintainer Massimo Aria <massimo.aria@gmail.com>

Repository CRAN

Date/Publication 2020-08-28 09:40:03 UTC

R topics documented:

altmetric	2
dsApi2df	2
dsApiRequest	4
dsAuth	5
dsQueryBuild	6
Index	8

`altmetric`*Gather altmetric metadata from a DOI*

Description

It gathers altmetric metadata from a DOI using Altmetric API (www.altmetric.com). A single research output may live online in multiple websites and can be talked about across dozens of different platforms.

Altmetric is a search engine which collects and collates all of this disparate information to obtain an informative view of the online activity surrounding your scholarly content.

Usage

```
altmetric(doi = "10.1016/j.joi.2017.08.007")
```

Arguments

`doi` is a character. It contains a list of DOIs. A DOI is a persistent identifier of a scholarly document.

Value

a data frame. Each row contains the full metadata record for each scholarly document.

For more extensive information about Altmetric, please visit: <https://www.altmetric.com>

Examples

```
## Not run:  
doi = "10.1016/j.joi.2017.08.007"  
  
df <- altmetric(doi = doi)  
  
## End(Not run)
```

`dsApi2df`*Convert json dimensions bibliographic data into a dataframe*

Description

It converts dimensions data, downloaded using DSL API, into a dataframe

Usage

```
dsApi2df(P, format = "bibliometrix")
```

Arguments

P is a list in json dimensions structure downloaded using the function dsApiRequest.

format is a character. If format = "bibliometrix" data will be converted in the bibliometrix complatible data format. If format = "raw" data will save in a data frame without any other data editing procedure.

Value

a dataframe containing bibliographic records or grants information.

To obtain a free access to Dimenions API fro no commercial use, please visit: <https://ds.digital-science.com/NoCostAgreement>

For more extensive information about dimensions API, please visit: <https://www.dimensions.ai/dimensions-apis/>

For more extensive information about bibliometrix R packagee, please visit: <https://www.bibliometrix.org>

See Also

[dsApiRequest](#)

[dsAuth](#)

[dsQueryBuild](#)

Examples

```
# Example 1: Querying a collection of publications

## Not run:
token <- dsAuth(username = "my.email@my.domain", password = "mypassword")
query <- dsQueryBuild(item = "publications", words = "bibliometric*",
                      type = "article", categories = "management",
                      start_year=1980,end_year = 2020)
D <- dsApiRequest(token = token, query = query, limit = 50000)
M <- dsApi2df(D)

## End(Not run)

# Example 2: Querying a collection of grants

## Not run:
token <- dsAuth(username = "my.email@my.domain", password = "mypassword")
query <- dsQueryBuild(item = "grants", words = "bibliometric*",
                      type = "", categories = "management",
                      start_year=1980,end_year = 2020)
D <- dsApiRequest(token = token, query = query, limit = 50000)
M <- dsApi2df(D)

## End(Not run)
```

`dsApiRequest`*Gather bibliographic records using Digital Science Dimensions API*

Description

It gathers bibliographic records from Digital Science Dimensions. The function `dsApiRequest` queries Dimensions using a DSL query formulated through the function `dsQueryBuilder`.

Usage

```
dsApiRequest(  
  token,  
  endpoint = "https://app.dimensions.ai/api/dsl.json",  
  query,  
  step = 100,  
  limit = 50000,  
  verbose = FALSE  
)
```

Arguments

<code>token</code>	is a character. It contains a valid token to query Dimensions database through DSL API. The token can be obtain using the function <code>dsAuth</code> with valid credentials (account and password) .
<code>endpoint</code>	is a character. It contains the endpoint url of Dimensions API. Default is <code>endpoint = "https://app.dimensions.ai/api/dsl.json"</code> .
<code>query</code>	is a character. It contains a search query formulated using the DSL API language. A query can be automatically generated using the function <code>dsQueryBuilder</code> .
<code>step</code>	is integer. It indicates the number of records to download at each API request. Default is <code>step = 100</code> .
<code>limit</code>	is integer. It indicates the max number of records to download. <code>limit</code> cannot be higher than 50.000 (as stated by Dimensions rules).
<code>verbose</code>	is logical.

Value

a list containing bibliographic metadata downloaded from Dimensions.

To obtain a free access to Dimensions API for no commercial use, please visit: <https://ds.digital-science.com/NoCostAgreement>

For more extensive information about dimensions API, please visit: <https://www.dimensions.ai/dimensions-apis/>

See Also[dsQueryBuild](#)[dsAuth](#)[dsApi2df](#)**Examples**

```
## Not run:
token <- dsAuth(username = "my.email@my.domain", password = "mypassword")
query <- dsQueryBuild(item = "publications", words = "bibliometric*",
                      type = "article", categories = "management",
                      start_year=1980,end_year = 2020)
D <- dsApiRequest(token = token, query = query, limit = 50000)

## End(Not run)
```

dsAuth*Obtain an API token from dimensions.ai*

Description

It generates a token request to dimensions.ai using account and password.

Usage

```
dsAuth(
  username,
  password,
  auth_endpoint = "https://app.dimensions.ai/api/auth.json",
  verbose = FALSE
)
```

Arguments

username	is a character.
password	is a character.
auth_endpoint	is a character. It contains the authentication endpoint url of Dimensions. Default is auth_endpoint = "https://app.dimensions.ai/api/auth.json"
verbose	is logical.

Value

a character containing an token o use dimensions API.

To obtain a free access to Dimenions API fro no commercial use, please visit: <https://ds.digital-science.com/NoCostAgreement>

For more extensive information about Dimensions API, please visit: <https://www.dimensions.ai/dimensions-apis/>

See Also

[dsApiRequest](#)

[dsQueryBuild](#)

[dsApi2df](#)

Examples

```
## Not run:  
token <- dsAuth(username = "my.email@my.domain", password = "mypassword")  
  
## End(Not run)
```

dsQueryBuild	<i>Generate a DSL query from a set of parameters It generates a valid query, written following the Dimensions Search Language (DSL), from a set of search parameters.</i>
--------------	---

Description

Generate a DSL query from a set of parameters It generates a valid query, written following the Dimensions Search Language (DSL), from a set of search parameters.

Usage

```
dsQueryBuild(  
  item = "publications",  
  words = "bibliometric*"  
  words_boolean_op = "OR",  
  full.search = FALSE,  
  type = "article",  
  categories = "",  
  output_fields = "all",  
  start_year = NULL,  
  end_year = NULL  
)
```

Arguments

item	is a character. It indicates the type of document to search. The argument can be equal to item = ("publications", "grants", "patents", "clinical_trials", "policy_documents". Default value is item = "publications".
words	is a character vector. It contains the search terms.
words_boolean_op	is character. It indicates which boolean operator have to be used to link words. It can be c("OR", "AND"). Default is "OR".
full.search	is logical. If TRUE, full-text search finds all instances of a term (keyword) in a document, or group of documents. If False, the search finds all instances in titles and abstracts only.
type	is a character. It indicates the document type to include in the search. Default is type = "article".
categories	is a character vector. It indicates the research categories to include in the search. If empty categories = "", all categories will be included in the search.
output_fields	is a character vector. It contains a list of fields which have to exported. Default is "all".
start_year	is integer. It indicate the starting publication year of the search timespan.
end_year	is integer. It indicate the ending publication year of the search timespan.

Value

a character containing the query in DSL format.

For more extensive information about Dimensions Search Language (DSL), please visit: <https://docs.dimensions.ai/dsl/>

To obtain a free access to Dimenions API fro no commercial use, please visit: <https://ds.digital-science.com/NoCostAgreement>

See Also

[dsApiRequest](#)

[dsAuth](#)

[dsApi2df](#)

Examples

```
## Not run:
query <- dsQueryBuild(item = "publications", words = "bibliometric*",
                      type = "article", categories = "management",
                      start_year=1980,end_year = 2020)

## End(Not run)
```

Index

altmetric, 2

dsApi2df, 2, 5–7

dsApiRequest, 3, 4, 6, 7

dsAuth, 3, 5, 5, 7

dsQueryBuild, 3, 5, 6, 6