

Package ‘puremoe’

October 15, 2024

Type Package

Title Pubmed Unified REtrieval for Multi-Output Exploration

Version 1.0.2

Maintainer Jason Timm <JaTimm@salud.unm.edu>

Description Access a variety of 'PubMed' data through a single, user-friendly interface, including abstracts <<https://pubmed.ncbi.nlm.nih.gov/>>, bibliometrics from 'iCite' <<https://icite.od.nih.gov/>>, pubtations from 'PubTator3' <<https://www.ncbi.nlm.nih.gov/research/pubtator3/>>, and full-text records from 'PMC' <<https://www.ncbi.nlm.nih.gov/pmc/>>.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 3.5)

Imports rentrez, textshape, xml2, data.table, httr, pbapply, jsonlite, rappdirs

Suggests knitr, rmarkdown

RoxygenNote 7.3.1

URL <https://github.com/jaytimm/puremoe>,
<https://jaytimm.github.io/puremoe/>

BugReports <https://github.com/jaytimm/puremoe/issues>

NeedsCompilation no

Author Jason Timm [aut, cre]

Repository CRAN

Date/Publication 2024-10-15 18:30:02 UTC

Contents

data_mesh_embeddings	2
data_mesh_thesaurus	3

data_mesh_trees	4
data_pharm_action	5
data_pmc_list	6
get_records	7
search_pubmed	8

Index	9
--------------	----------

data_mesh_embeddings *Download and Process 'MeSH' and 'SCR' Embeddings*

Description

This function downloads 'MeSH' and 'SCR' embeddings data from the specified URLs and processes it for use. The data is saved locally in RDS format. If the files do not exist, they will be downloaded and processed.

Usage

```
data_mesh_embeddings(
  path = NULL,
  use_persistent_storage = FALSE,
  force_install = FALSE
)
```

Arguments

path	A character string specifying the directory path where data should be stored. If not provided and persistent storage is requested, it defaults to a system-appropriate persistent location managed by 'rappdirs'.
use_persistent_storage	A logical value indicating whether to use persistent storage. If TRUE and no path is provided, data will be stored in a system-appropriate location. Defaults to FALSE, using a temporary directory.
force_install	A logical value indicating whether to force re-downloading of the data even if it already exists locally.

Details

This dataset is not viewable until it has been downloaded.

Citation

Noh, J., & Kavuluru, R. (2021). Improved biomedical word embeddings in the transformer era. *Journal of biomedical informatics*, 120, 103867.

Value

A data frame containing the processed Mesh and SCR embeddings data.

Examples

```
if (interactive()) {  
  data <- data_mesh_embeddings()  
}
```

data_mesh_thesaurus *Download and Combine 'MeSH' and Supplemental Thesauruses*

Description

This function downloads and combines the 'MeSH' (Medical Subject Headings) Thesaurus and a supplemental concept thesaurus. The data is sourced from specified URLs and stored locally for subsequent use. By default, the data is stored in a temporary directory. Users can opt into persistent storage by setting 'use_persistent_storage' to TRUE and optionally specifying a path.

Usage

```
data_mesh_thesaurus(  
  path = NULL,  
  use_persistent_storage = FALSE,  
  force_install = FALSE  
)
```

Arguments

path	A character string specifying the directory path where data should be stored. If not provided and persistent storage is requested, it defaults to a system-appropriate persistent location managed by 'rappdirs'.
use_persistent_storage	A logical value indicating whether to use persistent storage. If TRUE and no path is provided, data will be stored in a system-appropriate location. Defaults to FALSE, using a temporary directory.
force_install	A logical value indicating whether to force re-downloading of the data even if it already exists locally.

Value

A data.table containing the combined MeSH and supplemental thesaurus data.

Examples

```
if (interactive()) {  
  data <- data_mesh_thesaurus()  
}
```

`data_mesh_trees`*Download and Load 'MeSH' Trees Data*

Description

This function downloads and loads the 'MeSH' (Medical Subject Headings) Trees data.

Usage

```
data_mesh_trees(  
  path = NULL,  
  use_persistent_storage = FALSE,  
  force_install = FALSE  
)
```

Arguments

<code>path</code>	A character string specifying the directory path where data should be stored. If not provided and persistent storage is requested, it defaults to a system-appropriate persistent location managed by 'rappdirs'.
<code>use_persistent_storage</code>	A logical value indicating whether to use persistent storage. If TRUE and no path is provided, data will be stored in a system-appropriate location. Defaults to FALSE, using a temporary directory.
<code>force_install</code>	A logical value indicating whether to force re-downloading of the data even if it already exists locally.

Details

The data is sourced from specified URLs and stored locally for subsequent use. By default, the data is stored in a temporary directory. Users can opt into persistent storage by setting 'use_persistent_storage' to TRUE and optionally specifying a path.

Value

A data frame containing the MeSH Trees data.

Examples

```
data <- data_mesh_trees()
```

data_pharm_action	<i>Download and Load Pharmacological Actions Data</i>
-------------------	---

Description

This function downloads and loads pharmacological actions data from a specified URL.

Usage

```
data_pharm_action(  
  path = NULL,  
  use_persistent_storage = FALSE,  
  force_install = FALSE  
)
```

Arguments

path	A character string specifying the directory path where data should be stored. If not provided and persistent storage is requested, it defaults to a system-appropriate persistent location managed by 'rappdirs'.
use_persistent_storage	A logical value indicating whether to use persistent storage. If TRUE and no path is provided, data will be stored in a system-appropriate location. Defaults to FALSE, using a temporary directory.
force_install	A logical value indicating whether to force re-downloading of the data even if it already exists locally.

Details

The data is sourced from specified URLs and stored locally for subsequent use. By default, the data is stored in a temporary directory. Users can opt into persistent storage by setting 'use_persistent_storage' to TRUE and optionally specifying a path.

Value

A data frame containing pharmacological actions data.

Examples

```
data <- data_pharm_action()
```

data_pmc_list	<i>Download and Process 'PMC Open Access' File List</i>
---------------	---

Description

This function downloads the 'PubMed Central' (PMC) open access file list from the 'National Center for Biotechnology Information' (NCBI) and processes it for use.

Usage

```
data_pmc_list(  
  path = NULL,  
  use_persistent_storage = FALSE,  
  force_install = FALSE,  
  timeout = 300  
)
```

Arguments

path	A character string specifying the directory path where data should be stored. If not provided and persistent storage is requested, it defaults to a system-appropriate persistent location managed by 'rappdirs'.
use_persistent_storage	A logical value indicating whether to use persistent storage. If TRUE and no path is provided, data will be stored in a system-appropriate location. Defaults to FALSE, using a temporary directory.
force_install	A logical value indicating whether to force re-downloading of the data even if it already exists locally.
timeout	An integer indicating the timeout in seconds for the download. Defaults to 300 seconds.

Details

The data is sourced from the specified URL and stored locally for subsequent use. By default, the data is stored in a temporary directory. Users can opt into persistent storage by setting 'use_persistent_storage' to TRUE and optionally specifying a path.

Value

A data frame containing the processed PMC open access file list.

Examples

```
if (interactive()) {  
  data <- data_pmc_list()  
}
```

get_records	<i>Retrieve Data from 'NLM'/'PubMed' databases Based on PMIDs</i>
-------------	---

Description

This function retrieves different types of data (like 'PubMed' records, affiliations, 'iCites' data, etc.) from 'PubMed' based on provided PMIDs. It supports parallel processing for efficiency.

Usage

```
get_records(  
  pmids,  
  endpoint = c("pubtations", "icites", "pubmed_affiliations", "pubmed_abstracts",  
              "pmc_fulltext"),  
  cores = 3,  
  sleep = 1,  
  ncbi_key = NULL  
)
```

Arguments

pmids	A vector of PMIDs for which data is to be retrieved.
endpoint	A character vector specifying the type of data to retrieve ('pubtations', 'icites', 'affiliations', 'pubmed', 'pmc').
cores	Number of cores to use for parallel processing (default is 3).
sleep	Duration (in seconds) to pause after each batch
ncbi_key	(Optional) NCBI API key for authenticated access.

Value

A data.table containing combined results from the specified endpoint.

Examples

```
pmids <- c("38136652")  
results <- get_records(pmids, endpoint = "pubmed_abstracts", cores = 1)
```

`search_pubmed`*Search 'PubMed' Records*

Description

Performs a 'PubMed' search based on a query, optionally filtered by publication years. Returns a unique set of 'PubMed' IDs matching the query.

Usage

```
search_pubmed(  
  x,  
  start_year = NULL,  
  end_year = NULL,  
  retmax = 9999,  
  use_pub_years = TRUE  
)
```

Arguments

<code>x</code>	Character string, the search query.
<code>start_year</code>	Integer, the start year of publication date range (used if 'use_pub_years' is TRUE).
<code>end_year</code>	Integer, the end year of publication date range (used if 'use_pub_years' is TRUE).
<code>retmax</code>	Integer, maximum number of records to retrieve, defaults to 9999.
<code>use_pub_years</code>	Logical, whether to filter search by publication years, defaults to TRUE.

Value

Numeric vector of unique PubMed IDs.

Examples

```
ethnob1 <- search_pubmed("ethnobotany", 2010, 2012)
```


Index

`data_mesh_embeddings`, [2](#)
`data_mesh_thesaurus`, [3](#)
`data_mesh_trees`, [4](#)
`data_pharm_action`, [5](#)
`data_pmc_list`, [6](#)

`get_records`, [7](#)

`search_pubmed`, [8](#)