Package 'dhlabR'

June 9, 2023

Title National Library of Norway Quantitative Text Data API Tools

Version 1.0.2

Description
Tools for accessing data from National Library of Norway's dhlab (digital humanities laboratory).
Provides wrappers for accessing our API services at <https://api.nb.no/dhlab/>.
To learn more about dhlab, visit out site <https://www.nb.no/dh-lab/>.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.2.3

Imports dplyr, httr, jsonlite, purrr, tibble, zoo

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Lars Tungland [aut, cre],
Andre Kåsen [aut, cph]

Maintainer Lars Tungland <lars.tungland@nb.no>

Repository CRAN

Date/Publication 2023-06-09 14:10:11 UTC

R topics documented:

`get_collocations` *Get collocations for word in corpus*

Description

This function retrieves collocation data from a corpus using a given word and a list of unique identifiers (pids) of corpus data frame.

Usage

```
get_collocations(pids, word, before = 10, after = 10, sample_size = 5000)
```

Arguments

`pids`	A vector or data frame containing the unique identifiers of the texts in the corpus.
`word`	The target word for which you want to find concordances.
`before`	The number of words before the target word to include in the context (default is 10).
`after`	The number of words after the target word to include in the context (default is 10).
`sample_size`	The number of samples to retrieve from the API (default is 5000).

Value

A data frame of concordances.

Examples

```
pids <- c("URN:NBN:no-nb_digibok_2008051404065", "URN:NBN:no-nb_digibok_2010092120011")
word <- "."
collocations <- get_collocations(pids, word)
```

`get_concordance` *Retrieve Concordance for Words in Documents*

Description

This function obtains the concordance for specified words within given documents.

Usage

```
get_concordance(pids, words, window = 20, limit = 5000)
```

Arguments

pids	A vector or data frame containing document IDs.
words	A string of words (tokens) for which the concordance will be retrieved. For multiple tokens use keyword OR
window	An optional numeric value specifying the number of characters before and after the matching word (default is 20).
limit	An optional numeric value specifying the maximum number of results to return (default is 5000).

Value

A data frame containing the concordance results for each word in the specified documents. Returns NULL if the API request fails or no results are found.

Examples

```
document_ids <- c("URN:NBN:no-nb_digibok_2008051404065", "URN:NBN:no-nb_digibok_2010092120011")
tokens <- "Norge"
window <- 20
limit <- 1000
result <- get_concordance(document_ids, tokens, window, limit)
```

---

get_dispersion *Dispersion of tokens in a text*

Description

This function wraps a call to the dispersion service, which calculates the dispersion of a list of tokens throughout a text in the National Library of Norway's collection, given by the URN. The text is divided into chunks, and the count of tokens in each chunk is returned.

Usage

```
get_dispersion(urn = NULL, words = list(".", ","), window = 500, pr = 100)
```

Arguments

urn	A National Library of Norway URN to a text object.
words	A list or vector of words (tokens) to analyze for dispersion.
window	The size of the text chunk to count the tokens within.
pr	(Per) Determines the step size for moving forward to the next chunk. If 'pr' is equal to 'window', the text is divided into non-overlapping chunks of size 'window'. If 'pr' is smaller than 'window', the chunks will overlap, creating a smoother curve.

Value

A data frame with the count of tokens in each chunk.

Examples

```
urn <- "URN:NBN:no-nb_digibok_2013060406055"
words <- c("Dracula", "Mina", "Helsing")
window <- 1000
pr <- 1000
dispersion_result <- get_dispersion(urn, words, window, pr)
```

---

get_document_corpus *Get Document Corpus*

Description

Retrieve a corpus of documents based on the given parameters.

Usage

```
get_document_corpus(
  doctype = "digibok",
  author = NULL,
  ddk = NULL,
  freetext = NULL,
  subject = NULL,
  from_timestamp = NULL,
  to_timestamp = NULL,
  publisher = NULL,
  limit = 10,
  order_and_limit_by_rank = NULL,
  title = NULL,
  from_year = NULL,
  to_year = NULL,
  fulltext = NULL,
  lang = "nob"
)
```

Arguments

doctype	Character, document type (default: 'digibok')
author	Character, author name (default: NULL)
ddk	Character, Dewey Decimal Classification (default: NULL)
freetext	Character, free text search (default: NULL)
subject	Character, subject of the document (default: NULL)
from_timestamp	Character, timestamp range start (default: NULL)

to_timestamp	Character, timestamp range end (default: NULL)
publisher	Character, publisher name (default: NULL)
limit	Integer, maximum number of results (default: 10)
order_and_limit_by_rank	Logical, order and limit results by rank (default: NULL)
title	Character, title of the document (default: NULL)
from_year	Integer, year range start (default: NULL)
to_year	Integer, year range end (default: NULL)
fulltext	Character, full text search (default: NULL)
lang	Character, language code (default: 'nob')

Value

A data frame of metadata

Examples

```
get_document_corpus(doctype = 'digibok', author = 'Henrik Ibsen', limit = 2)
```

---

get_document_frequencies

Retrieve Token Frequencies in Documents

Description

This function obtains token frequencies within specified documents.

Usage

```
get_document_frequencies(pids, cutoff = 0, words = NULL)
```

Arguments

pids	A vector or data frame containing document IDs.
cutoff	A numeric value specifying the frequency cutoff for tokens.
words	A vector of words (tokens) to retrieve frequencies for.

Value

A list containing the following elements for each document:

- Document ID
- Token
- Token frequency in the document
- Total tokens in the document

Examples

```
document_ids <- c("URN:NBN:no-nb_digibok_2008051404065", "URN:NBN:no-nb_digibok_2010092120011")
frequency_cutoff <- 10
tokens <- c(".", ",", "men")
result <- get_document_frequencies(document_ids, frequency_cutoff, tokens)
```

---

get_metadata *Get National Library Metadata for identifiers*

Description

This function retrieves metadata for objects from the National Library API based on either a vector of dhlabids or a vector of National Library URNs.

Usage

```
get_metadata(dhlabids = NULL, urns = NULL)
```

Arguments

dhlabids	A vector of dhlabids (default is NULL). When provided, the function will use dhlabids to fetch metadata.
urns	A vector of National Library URNs (default is NULL). When provided, the function will use URNs to fetch metadata.

Value

A dataframe containing the National Library metadata for the specified objects.

Examples

```
urns_example <- c("URN:NBN:no-nb_digibok_2008051404065", "URN:NBN:no-nb_digibok_2010092120011")
metadata_urns <- get_metadata(urns = urns_example)
```

---

get_ngram_from_books *Get Ngram Count per Year for National Library Book Collection*

Description

This function queries the National Library's book collection API to retrieve the ngram count per year for the specified parameters. It can be used to plot an ngram based on the words' presence in books in the library's collection.

Usage

```
get_ngram_from_books(
  city = NULL,
  ddk = NULL,
  lang = NULL,
  period = list(),
  publisher = NULL,
  title = NULL,
  topic = NULL,
  word = list("hus", "blokk")
)
```

Arguments

city	(character, optional) The city of publication. Default is NULL.
ddk	(character, optional) The Dewey Decimal Classification (DDC) code. Default is NULL.
lang	(character, optional) The language code of the books. Default is NULL.
period	(list, optional) A list containing the start and end years of the period to search. Default is an empty list.
publisher	(character, optional) The publisher's name. Default is NULL.
title	(character, optional) The title or a part of the title of the books. Default is NULL.
topic	(character, optional) A topic or subject associated with the books. Default is NULL.
word	(list, optional) A list of words (ngrams) to search for in the books. Default is list("hus", "blokk").

Value

A data frame with the ngram count per year for the specified parameters.

Examples

```
# Get ngram count for the words "hus" and "blokk" in the specified period
get_ngram_from_books(period = list(1990, 2000))

# Get ngram count for the word "library" in English books
get_ngram_from_books(lang = "eng", word = list("library"))
```

`get_ngram_from_newspapers` *Get Ngram Count per Year for National Library Newspaper Collection*

Description

This function queries the National Library's book collection API to retrieve the ngram count per year for the specified parameters. It can be used to plot an ngram based on the words' presence in books in the library's collection.

Usage

```
get_ngram_from_newspapers(
  city = NULL,
  ddk = NULL,
  lang = NULL,
  period = list(),
  publisher = NULL,
  title = NULL,
  topic = NULL,
  word = list("hus", "blokk")
)
```

Arguments

`city`	(character, optional) The city of publication. Default is NULL.
`ddk`	(character, optional) The Dewey Decimal Classification (DDC) code. Default is NULL.
`lang`	(character, optional) The language code of the books. Default is NULL.
`period`	(list, optional) A list containing the start and end years of the period to search. Default is an empty list.
`publisher`	(character, optional) The publisher's name. Default is NULL.
`title`	(character, optional) The title or a part of the title of the books. Default is NULL.
`topic`	(character, optional) A topic or subject associated with the books. Default is NULL.
`word`	(list, optional) A list of words (ngrams) to search for in the books. Default is list("hus", "blokk").

Value

A data frame with the ngram count per year for the specified parameters.

`get_reference_words` *Retrieve Reference Words Count and Relative Frequency*

Description

This function obtains the count and relative frequency of a vector of words within a year range for specified document types.

Usage

```
get_reference_words(
  doctype = "digibok",
  from_year = 1990,
  to_year = 2000,
  words = NULL
)
```

Arguments

`doctype`	A character string indicating the document type. One of "digibok", "digavis", or "digitidsskrift".
`from_year`	A numeric value indicating the starting year of the range.
`to_year`	A numeric value indicating the ending year of the range.
`words`	A vector of words for which the count and relative frequency will be retrieved.

Value

A list containing the count and relative frequency of the specified words within the given year range and document type.

Examples

```
doctype <- "digibok"
from_year <- 1900
to_year <- 2000
words <- c("og", "eller", "men")
result <- get_reference_words(doctype, from_year, to_year, words)
```

get_urn_frequencies *Get word count frequencies for a list of URNs or dhlabids*

Description

This function takes a list of National Library of Norway (NB) identifiers, either URNs or dhlabids, and returns the word count for each object. It queries the National Library's API to fetch the word count data.

Usage

```
get_urn_frequencies(urns = NULL, dhlabids = NULL)
```

Arguments

urns	A list or data frame of URNs from the National Library of Norway. If a data frame, it should have a column named 'urn'.
dhlabids	A list of 'dhlabid' ids from National Library DHLAB.

Value

A data frame with two columns: 'dhlabid' and 'frequencies'. Each row represents a library text resource with its corresponding word count.

Examples

```
# Example usage with a list of URNs
urn_list <- c("URN:NBN:no-nb_digibok_2008051404065", "URN:NBN:no-nb_digibok_2010092120011")
word_counts <- get_urn_frequencies(urn_list)
print(word_counts)

# Example usage with a data frame of URNs
urn_list <- c("URN:NBN:no-nb_digibok_2008051404065", "URN:NBN:no-nb_digibok_2010092120011")
urn_dataframe <- data.frame(urn = urn_list)
word_counts <- get_urn_frequencies(urn_dataframe)
```

ngram *Function to get and convert NGRAM*

Description

Function to get and convert NGRAM

Usage

```
ngram(
  word = "havet",
  corpus = "bok",
  language = NULL,
  smooth = 1,
  years = c(1810, 2013),
  mode = "relative"
)
```

Arguments

word	The word to get NGRAM for. Default is "havet".
corpus	The corpus to use. Options are 'avis' and 'bok'. Default is "bok".
language	The language to use. Default is NULL.
smooth	Smoothing factor. Default is 1.
years	A vector that contains the start and end years. Default is c(1810,2013).
mode	The mode to use. Default is 'relative'.

Value

A data frame that contains the NGRAM.

Index