Package 'specieshindex'

April 9, 2021

Type Package

Title How (scientifically) popular is a given species?

Version 0.1.1

Date 2021-01-19

Author Jessica Tam

Maintainer Jessica Tam <tamtinying@gmail.com>

Description Finds the h-index of a species.

Depends R (¿= 3.5.0)

LazyData true

Imports rscopus,
wosr,
rbace,
lens2r,
taxize,
dplyr,
data.table,
httr,
XML,
jsonlite

Suggests testthat,
knitr,
roxygen2,
devtools,
rmarkdown,
ggplot2,
RefManageR

License MIT + file LICENSE

URL https://github.com/jessicatytam/specieshindex

RoxygenNote 7.1.1

VignetteBuilder knitr

Encoding UTF-8

R topics documented:

`Allindices` *Index summary*

Description

This function returns a dataframe of the summary of all of the indices.

Usage

```
Allindices(data, genus, species)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.
genus	Genus classification from the binomial name.
species	Species classification from the binomial name.

Value

A datarame of all of the indices in the package.

Examples

```
data(Woylie)
Allindices(Woylie, genus = "genus_name", species = "species_name")
```

ARRatio *Article:Review ratio*

Description

This function calculates the percentage ratio of article:rerview.

Usage

```
ARRatio(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

A character value of the percentage ratio of the number of articles and reviews.

Examples

```
data(Woylie)
ARRatio(Woylie)
```

CountSpT *Search count from Scopus - title only*

Description

This function counts the total number of search results. It counts the publications with the binomial name in the title only. A check will be conducted via gnr_resolve to validate the genus and species names.

Usage

```
CountSpT(
  genus,
  species,
  synonyms,
  additionalkeywords,
  APIkey,
  datatype = "application/xml"
)
```

Arguments

genus	Genus classification from the binomial name.
species	Species classification from the binomial name.
synonyms	Alternate species names.
additionalkeywords	Optional search terms.
APIkey	Scopus API key needed to access and download data from their database.
datatype	Formats the URL to be sent to the API. The default is "application/xml".

Value

Search count of the species with the given genus and species.

References

Chamberlain, S. & Szocs, E. (2013). taxize - taxonomic search and retrieval in R. *F1000Research, 2*, 191.

Examples

```
## Not run:
CountSpT("Bettongia", "penicillata", APIkey = "myAPI")

#lower case letter in genus is also accepted and will return identical results

CountSpT("bettongia", "penicillata", APIkey = "myAPI")
```

```
## End(Not run)
## Not run:
CountSpT("Bettongia", "penicillata", "conserv*", "myAPI")

#lower case letter in genus is also accepted and will return identical results

CountSpT("bettongia", "penicillata", "conserv*", "myAPI")

## End(Not run)
```

`CountSpTAK` *Search count from Scopus - title, abstract, and keywords*

Description

This function counts the total number of search results. It counts the publications with the binomial name in the title, abstract, and keywords. A check will be conducted via `gnr_resolve` to validate the genus and species names.

Usage

```
CountSpTAK(
  genus,
  species,
  synonyms,
  additionalkeywords,
  APIkey,
  datatype = "application/xml"
)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.
`APIkey`	Scopus API key needed to access and download data from their database.
`datatype`	Formats the URL to be sent to the API. The default is "application/xml".

Value

Search count of the species with the given `genus` and `species`.

References

Chamberlain, S. & Szocs, E. (2013). taxize - taxonomic search and retrieval in R. *F1000Research, 2*, 191.

Examples

```
## Not run:
CountSpTAK("Bettongia", "penicillata", APIkey = "myAPI")

#lower case letter in genus is also accepted and will return identical results

CountSpTAK("bettongia", "penicillata", APIkey = "myAPI")

## End(Not run)
## Not run:
CountSpTAK("Bettongia", "penicillata", "conserv*", "myAPI")

#lower case letter in genus is also accepted and will return identical results

CountSpTAK("bettongia", "penicillata", "conserv*", "myAPI")

## End(Not run)
```

`CountSpTAK_lens` *Search count from Lens - title, abstract and author keywords*

Description

This function counts the total number of search results. It counts the publications with the binomial name in the title, abstract and author keywords. A check will be conducted via `gnr_resolve` to validate the genus and species names.

Usage

```
CountSpTAK_lens(genus, species, synonyms, additionalkeywords, token)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.
`token`	Lens token needed to access and download data from their database.

Value

Search count of the species with the given `genus` and `species`.

References

Chamberlain, S. & Szocs, E. (2013). taxize - taxonomic search and retrieval in R. *F1000Research, 2*, 191.

Examples

```
## Not run:
CountSpTAK_lens("Bettongia", "penicillata")

#lower case letter in genus is also accepted and will return identical results

CountSpTAK_lens("bettongia", "penicillata")

## End(Not run)
## Not run:
CountSpTAK_lens("Bettongia", "penicillata", "conserv*")

#lower case letter in genus is also accepted and will return identical results

CountSpTAK_lens("bettongia", "penicillata", "conserv*")

## End(Not run)
```

`CountSpTAK_wos` *Search count from Web of Science - title, abstract and author keywords*

Description

This function counts the total number of search results. It counts the publications with the binomial name in the title, abstract and author keywords. A check will be conducted via `gnr_resolve` to validate the genus and species names.

Usage

```
CountSpTAK_wos(genus, species, synonyms, additionalkeywords)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.

Value

Search count of the species with the given `genus` and `species`.

References

Chamberlain, S. & Szocs, E. (2013). taxize - taxonomic search and retrieval in R. *F1000Research, 2*, 191.

Examples

```
## Not run:
CountSpTAK_wos("Bettongia", "penicillata")

#lower case letter in genus is also accepted and will return identical results

CountSpTAK_wos("bettongia", "penicillata")

## End(Not run)
## Not run:
CountSpTAK_wos("Bettongia", "penicillata", "conserv*")

#lower case letter in genus is also accepted and will return identical results

CountSpTAK_wos("bettongia", "penicillata", "conserv*")

## End(Not run)
```

`CountSpT_lens` *Search count from Lens - title only*

Description

This function counts the total number of search results. It counts the publications with the binomial name in the title only. A check will be conducted via `gnr_resolve` to validate the genus and species names.

Usage

```
CountSpT_lens(genus, species, synonyms, additionalkeywords, token)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.
`token`	Lens token needed to access and download data from their database.

Value

Search count of the species with the given `genus` and `species`.

References

Chamberlain, S. & Szocs, E. (2013). taxize - taxonomic search and retrieval in R. *F1000Research, 2*, 191.

Examples

```
## Not run:
CountSpT_lens("Bettongia", "penicillata")

#lower case letter in genus is also accepted and will return identical results

CountSpT_lens("bettongia", "penicillata")

## End(Not run)
## Not run:
CountSpT_lens("Bettongia", "penicillata", "conserv*")

#lower case letter in genus is also accepted and will return identical results

CountSpT_lens("bettongia", "penicillata", "conserv*")

## End(Not run)
```

`CountSpT_wos` *Search count from Web of Science - title only*

Description

This function counts the total number of search results. It counts the publications with the binomial name in the title only. A check will be conducted via `gnr_resolve` to validate the genus and species names.

Usage

```
CountSpT_wos(genus, species, synonyms, additionalkeywords)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.

Value

Search count of the species with the given `genus` and `species`.

References

Chamberlain, S. & Szocs, E. (2013). taxize - taxonomic search and retrieval in R. *F1000Research, 2*, 191.

Examples

```
## Not run:
CountSpT_wos("Bettongia", "penicillata")

#lower case letter in genus is also accepted and will return identical results

CountSpT_wos("bettongia", "penicillata")

## End(Not run)
## Not run:
CountSpT_wos("Bettongia", "penicillata", "conserv*")

#lower case letter in genus is also accepted and will return identical results

CountSpT_wos("bettongia", "penicillata", "conserv*")

## End(Not run)
```

`FetchSpT` *Fetch data from Scopus - title only*

Description

This function fetches citation information from Scopus using genus and species name found in the title of the publications. Duplicates are to be removed by the user after fetching the data.

Usage

```
FetchSpT(genus, species, synonyms, additionalkeywords, language = 0, APIkey)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.
`language`	Language of the paper; default is 0, enter 1 to retrieve the variable.
`APIkey`	Scopus API key needed to access and download data from their database.

Value

A dataframe of the species' citation records with the given `genus` and `species`.

Examples

```
## Not run:
FetchSpT("Bettongia", "penicillata", APIkey = "myAPI")

#lower case letter in genus is also accepted and will return identical results

FetchSpT("bettongia", "penicillata", APIkey = "myAPI")

## End(Not run)
## Not run:
FetchSpT("Bettongia", "penicillata", "conserv*", "myAPI")

#lower case letter in genus is also accepted and will return identical results

FetchSpT("bettongia", "penicillata", "conserv*", "myAPI")

## End(Not run)
```

`FetchSpTAK` *Fetch data from Scopus - title, abstract and keywords*

Description

This function fetches citation information from Scopus using genus and species name found in the title, abstract and keywords of the publications. Duplicates are to be removed by the user after fetching the data.

Usage

```
FetchSpTAK(genus, species, synonyms, additionalkeywords, language = 0, APIkey)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.
`language`	Language of the paper; default is 0, enter 1 to retrieve the variable.
`APIkey`	Scopus API key needed to access and download data from their database.

Value

A dataframe of the species' citation records with the given `genus` and `species`.

Examples

```
## Not run:
FetchSpTAK("Bettongia", "penicillata", APIkey = "myAPI")

#lower case letter in genus is also accepted and will return identical results

FetchSpTAK("bettongia", "penicillata", APIkey = "myAPI")

## End(Not run)
## Not run:
FetchSpTAK("Bettongia", "penicillata", "conserv*", "myAPI")

#lower case letter in genus is also accepted and will return identical results

FetchSpTAK("bettongia", "penicillata", "conserv*", "myAPI")

## End(Not run)
```

FetchSpTAK_lens *Fetch data from Lens - title, abstract, and keywords.*

Description

This function fetches citation information from Lens using genus and species name found in the title, abstract, and keywords of the publications. Duplicates are to be removed by the user after fetching the data.

Usage

```
FetchSpTAK_lens(genus, species, synonyms, additionalkeywords, token)
```

Arguments

genus	Genus classification from the binomial name.
species	Species classification from the binomial name.
synonyms	Alternate species names.
additionalkeywords	Optional search terms.
token	Lens token needed to access and download data from their database.

Value

A dataframe of the species' citation records with the given genus and species.

Examples

```
## Not run:
FetchSpTAK_lens("Bettongia", "penicillata", token = "mytoken")

#lower case letter in genus is also accepted and will return identical results

FetchSpTAK_lens("bettongia", "penicillata", token = "mytoken")

## End(Not run)
## Not run:
FetchSpTAK_lens("Bettongia", "penicillata", "conserv*", token = "mytoken")

#lower case letter in genus is also accepted and will return identical results

FetchSpTAK_lens("bettongia", "penicillata", "conserv*", token = "mytoken")

## End(Not run)
```

`FetchSpTAK_wos` *Fetch data from Web of Science - title, abstract and author keywords.*

Description

This function fetches citation information from Web of Science using genus and species name found in the title, abstract and author keywords of the publications. Duplicates are to be removed by the user after fetching the data.

Usage

```
FetchSpTAK_wos(genus, species, synonyms, additionalkeywords)
```

Arguments

`genus`	Genus classification from the binomial name.
`species`	Species classification from the binomial name.
`synonyms`	Alternate species names.
`additionalkeywords`	Optional search terms.

Value

A dataframe of the species' citation records with the given `genus` and `species`.

Examples

```
## Not run:
FetchSpTAK_wos("Bettongia", "penicillata")

#lower case letter in genus is also accepted and will return identical results

FetchSpTAK_wos("bettongia", "penicillata")

## End(Not run)
## Not run:
FetchSpTAK_wos("Bettongia", "penicillata", "conserv*")

#lower case letter in genus is also accepted and will return identical results

FetchSpTAK_wos("bettongia", "penicillata", "conserv*")

## End(Not run)
```

FetchSpT_lens *Fetch data from Lens - title only*

Description

This function fetches citation information from Lens using genus and species name found in the title of the publications. Duplicates are to be removed by the user after fetching the data.

Usage

```
FetchSpT_lens(genus, species, synonyms, additionalkeywords, token)
```

Arguments

genus	Genus classification from the binomial name.
species	Species classification from the binomial name.
synonyms	Alternate species names.
additionalkeywords	Optional search terms.
token	Lens token needed to access and download data from their database.

Value

A dataframe of the species' citation records with the given genus and species.

Examples

```
## Not run:
FetchSpT_lens("Bettongia", "penicillata", token = "mytoken")

#lower case letter in genus is also accepted and will return identical results

FetchSpT_lens("bettongia", "penicillata", token = "mytoken")

## End(Not run)
## Not run:
FetchSpT_lens("Bettongia", "penicillata", "conserv*", token = "mytoken")

#lower case letter in genus is also accepted and will return identical results

FetchSpT_lens("bettongia", "penicillata", "conserv*", token = "mytoken")

## End(Not run)
```

FetchSpT_wos *Fetch data from Web of Science - title only*

Description

This function fetches citation information from Web of Science using genus and species name found in the title of the publications. Duplicates are to be removed by the user after fetching the data.

Usage

```
FetchSpT_wos(genus, species, synonyms, additionalkeywords)
```

Arguments

genus	Genus classification from the binomial name.
species	Species classification from the binomial name.
synonyms	Alternate species names.
additionalkeywords	Optional search terms.

Value

A dataframe of the species' citation records with the given genus and species.

Examples

```
## Not run:
FetchSpT_wos("Bettongia", "penicillata")

#lower case letter in genus is also accepted and will return identical results

FetchSpT_wos("bettongia", "penicillata")

## End(Not run)
## Not run:
FetchSpT_wos("Bettongia", "penicillata", "conserv*")

#lower case letter in genus is also accepted and will return identical results

FetchSpT_wos("bettongia", "penicillata", "conserv*")

## End(Not run)
```

Koala *Koala dataset*

Description

Citation records of koala (Phascolarctos cinereus) from Scopus. Data was retrieved on 10 July 2020.

Usage

```
Koala
```

Format

A data frame with 773 rows and 20 variables

Source

http://api.elsevier.com/content/search/scopus

---

languages *Languages*

Description

List of languages of documents found on Scopus. Data was retrieved on 9 March 2021.

Usage

```
languages
```

Format

A csv file with the complete list of languages of documents found on Scopus.

Source

https://www.elsevier.com/solutions/scopus/how-scopus-works/content

---

`Platypus` *Platypus dataset*

Description

Citation records of platypus (Ornithorhynchus anatinus) from Scopus. Data was retrieved on 10 July 2020.

Usage

```
Platypus
```

Format

A data frame with 321 rows and 20 variables

Source

http://api.elsevier.com/content/search/scopus

`Quokka` *Quokka dataset*

Description

Citation records of quokka (Setonix brachyurus) from Scopus. Data was retrieved on 10 July 2020.

Usage

```
Quokka
```

Format

A data frame with 242 rows and 20 variables

Source

http://api.elsevier.com/content/search/scopus

SpH5 *Species h5 index*

Description

This function calculates the h-index of a species in the past 5 years.

Usage

```
SpH5(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

H5 index.

References

Suzuki, H. (2012). *Google Scholar Metrics for Publications.* Retrieved from https://scholar.googleblog.com/2012/04/google-scholar-metrics-for-publications.html.

Examples

```
data(Woylie)
SpH5(Woylie)
```

SpHAfterdate *Species h-index with a given time frame*

Description

This function calculates the h-index using a given date up till the newest record.

Usage

```
SpHAfterdate(data, date)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.
date	The lower limit of the timeframe.

Value

H-index of the given time period.

Examples

```
data(Woylie)
SpHAfterdate(Woylie, "2000-01-01")
```

`SpHindex` *Species h-index*

Description

This function calculates the h-index of a species.

Usage

```
SpHindex(data)
```

Arguments

`data`	The dataframe generated from `FetchSpT` or `FetchSpTAK`.

Value

H-index.

References

Bertoli-Barsotti, L. & Lando, T. (2015). On a formula for the h-index. *Journal of Informetrics, 9*(4), 762-776.
Hirsch, J. (2005). An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences of the United States of America, 102*(46), 16569-16572.

Examples

```
data(Woylie)
SpHindex(Woylie)
```

Spi10 *Species i10 index*

Description

This function calculates the i10 index of a species. i10 index counts all of the publications with 10 or more citations.

Usage

```
Spi10(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

i10 index.

References

Cornell University (2019). *i10-index.* Retrieved from https://guides.library.cornell.edu/c.php?g=32272&p=203393.

Examples

```
data(Woylie)
Spi10(Woylie)
```

SpMindex *Species m-index*

Description

This function calculates the m-index of species. M-index uses the h-index and divides it by the number of years of activity.

Usage

```
SpMindex(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

M-index.

References

University of Pittsburgh (2019). *Research Impact and Metrics: Author metrics.* Retrieved from https://pitt.libguides.com/bibliometricIndicators/AuthorMetrics.

Examples

```
data(Woylie)
SpMindex(Woylie)
```

`TotalArt` *Total Article*

Description

This function calculates the total number of articles.

Usage

```
TotalArt(data)
```

Arguments

`data`	The dataframe generated from `FetchSpT` or `FetchSpTAK`.

Value

An integer of the total number of articles.

Examples

```
data(Woylie)
TotalArt(Woylie)
```

TotalCite *Total citations*

Description

This function calculates the total number of citations.

Usage

```
TotalCite(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

A numerical value of the total number of citations.

Examples

```
data(Woylie)
TotalCite(Woylie)
```

TotalJournals *Total journals*

Description

This function calculates the total number of journals.

Usage

```
TotalJournals(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

An integer of the total number of journals.

Examples

```
data(Woylie)
TotalJournals(Woylie)
```

TotalPub *Total publications*

Description

This function calculates the total number of publications.

Usage

```
TotalPub(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

An integer of the total number of publications.

Examples

```
data(Woylie)
TotalPub(Woylie)
```

TotalRev *Total reviews*

Description

This function calculates the total number of reviews.

Usage

```
TotalRev(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

An integer of the total number of reviews.

Examples

```
data(Woylie)
TotalRev(Woylie)
```

Woylie *Woylie dataset*

Description

Citation records of woylie (Bettongia penicillata) from Scopus. Data was retrieved on 10 July 2020.

Usage

```
Woylie
```

Format

A data frame with 113 rows and 20 variables

Source

http://api.elsevier.com/content/search/scopus

YearsPublishing *Years since first publication*

Description

The number of years since the first publication in relation to the species.

Usage

```
YearsPublishing(data)
```

Arguments

data	The dataframe generated from FetchSpT or FetchSpTAK.

Value

Number of years.

Examples

```
data(Woylie)
YearsPublishing(Woylie)
```

Index