Package ‘wikitaxa’

October 12, 2022

Title Taxonomic Information from 'Wikipedia'

Description 'Taxonomic' information from 'Wikipedia', 'Wikicommons', 'Wikispecies', and 'Wikidata'. Functions included for getting taxonomic information from each of the sources just listed, as well performing taxonomic search.

Version 0.4.0

License MIT + file LICENSE

URL https://docs.ropensci.org/wikitaxa,
https://github.com/ropensci/wikitaxa

BugReports https://github.com/ropensci/wikitaxa/issues

LazyLoad yes

LazyData yes

Encoding UTF-8

Language en-US

VignetteBuilder knitr

Depends R(>= 3.2.1)

Imports WikidataR, data.table, curl, crul (>= 0.3.4), tibble, jsonlite, xml2

Suggests testthat, knitr, rmarkdown, vcr

RoxygenNote 7.1.0

X-schema.org-applicationCategory Taxonomy

X-schema.org-keywords taxonomy, species, API, web-services, Wikipedia, vernacular, Wikispecies, Wikicommons

X-schema.org-isPartOf https://ropensci.org

NeedsCompilation no

Author Scott Chamberlain [aut, cre],
Ethan Welty [aut]

Maintainer Scott Chamberlain <myrmecocystus+r@gmail.com>

Repository CRAN

Date/Publication 2020-06-29 15:30:03 UTC

R topics documented:

`wikitaxa-package` *wikitaxa*

Description

Taxonomic Information from Wikipedia

Author(s)

Scott Chamberlain <`myrmecocystus@gmail.com`>

Ethan Welty

`wikipedias` *List of Wikipedias*

Description

data.frame of 295 rows, with 3 columns:

- language - language
- language_local - language in local name
- wiki - langugae code for the wiki

Details

From `https://meta.wikimedia.org/wiki/List_of_Wikipedias`

wt_data *Wikidata taxonomy data*

Description

Wikidata taxonomy data

Usage

```
wt_data(x, property = NULL, ...)

wt_data_id(x, language = "en", limit = 10, ...)
```

Arguments

x	(character) a taxonomic name
property	(character) a property id, e.g., P486
...	curl options passed on to `httr::GET()`
language	(character) two letter language code
limit	(integer) records to return. Default: 10

Details

Note that `wt_data` can take a while to run since when fetching claims it has to do so one at a time for each claim

You can search things other than taxonomic names with `wt_data` if you like

Value

`wt_data` searches Wikidata, and returns a list with elements:

- labels - data.frame with columns: language, value
- descriptions - data.frame with columns: language, value
- aliases - data.frame with columns: language, value
- sitelinks - data.frame with columns: site, title
- claims - data.frame with columns: claims, property_value, property_description, value (comma separted values in string)

`wt_data_id` gets the Wikidata ID for the searched term, and returns the ID as character

Examples

```
## Not run:
# search by taxon name
# wt_data("Mimulus alsinoides")

# choose which properties to return
wt_data(x="Mimulus foliatus", property = c("P846", "P815"))

# get a taxonomic identifier
wt_data_id("Mimulus foliatus")
# the id can be passed directly to wt_data()
# wt_data(wt_data_id("Mimulus foliatus"))

## End(Not run)
```

---

wt_wikicommons *WikiCommons*

Description

WikiCommons

Usage

```
wt_wikicommons(name, utf8 = TRUE, ...)

wt_wikicommons_parse(
  page,
 types = c("langlinks", "iwlinks", "externallinks", "common_names", "classification"),
  tidy = FALSE
)

wt_wikicommons_search(query, limit = 10, offset = 0, utf8 = TRUE, ...)
```

Arguments

`name`	(character) Wiki name - as a page title, must be length 1
`utf8`	(logical) If `TRUE`, encodes most (but not all) non-ASCII characters as UTF-8 instead of replacing them with hexadecimal escape sequences. Default: `TRUE`
`...`	curl options, passed on to `httr::GET()`
`page`	(`httr::response()`) Result of `wt_wiki_page()`
`types`	(character) List of properties to parse
`tidy`	(logical). tidy output to data.frame's if possible. Default: `FALSE`
`query`	(character) query terms
`limit`	(integer) number of results to return. Default: 10
`offset`	(integer) record to start at. Default: 0

Value

wt_wikicommons returns a list, with slots:

- langlinks - language page links
- externallinks - external links
- common_names - a data.frame with name and language columns
- classification - a data.frame with rank and name columns

wt_wikicommons_parse returns a list

wt_wikicommons_search returns a list with slots for continue and query, where query holds the results, with query$search slot with the search results

References

https://www.mediawiki.org/wiki/API:Search for help on search

Examples

```
## Not run:
# high level
wt_wikicommons(name = "Malus domestica")
wt_wikicommons(name = "Pinus contorta")
wt_wikicommons(name = "Ursus americanus")
wt_wikicommons(name = "Balaenoptera musculus")

wt_wikicommons(name = "Category:Poeae")
wt_wikicommons(name = "Category:Pinaceae")

# low level
pg <- wt_wiki_page("https://commons.wikimedia.org/wiki/Malus_domestica")
wt_wikicommons_parse(pg)

# search wikicommons
# FIXME: utf=FALSE for now until curl::curl_escape fix
# https://github.com/jeroen/curl/issues/228
wt_wikicommons_search(query = "Pinus", utf8 = FALSE)

## use search results to dig into pages
res <- wt_wikicommons_search(query = "Pinus", utf8 = FALSE)
lapply(res$query$search$title[1:3], wt_wikicommons)

## End(Not run)
```

`wt_wikipedia` *Wikipedia*

Description

Wikipedia

Usage

```
wt_wikipedia(name, wiki = "en", utf8 = TRUE, ...)

wt_wikipedia_parse(
  page,
  types = c("langlinks", "iwlinks", "externallinks", "common_names", "classification"),
  tidy = FALSE
)

wt_wikipedia_search(
  query,
  wiki = "en",
  limit = 10,
  offset = 0,
  utf8 = TRUE,
  ...
)
```

Arguments

`name`	(character) Wiki name - as a page title, must be length 1
`wiki`	(character) wiki language. default: en. See wikipedias for language codes.
`utf8`	(logical) If `TRUE`, encodes most (but not all) non-ASCII characters as UTF-8 instead of replacing them with hexadecimal escape sequences. Default: `TRUE`
`...`	curl options, passed on to `httr::GET()`
`page`	(`httr::response()`) Result of `wt_wiki_page()`
`types`	(character) List of properties to parse
`tidy`	(logical). tidy output to data.frame's if possible. Default: `FALSE`
`query`	(character) query terms
`limit`	(integer) number of results to return. Default: 10
`offset`	(integer) record to start at. Default: 0

Value

wt_wikipedia returns a list, with slots:

- langlinks - language page links
- externallinks - external links
- common_names - a data.frame with name and language columns
- classification - a data.frame with rank and name columns
- synonyms - a character vector with taxonomic names

wt_wikipedia_parse returns a list with same slots determined by the types parmeter

wt_wikipedia_search returns a list with slots for continue and query, where query holds the results, with query$search slot with the search results

References

https://www.mediawiki.org/wiki/API:Search for help on search

Examples

```
## Not run:
# high level
wt_wikipedia(name = "Malus domestica")
wt_wikipedia(name = "Malus domestica", wiki = "fr")
wt_wikipedia(name = "Malus domestica", wiki = "da")

# low level
pg <- wt_wiki_page("https://en.wikipedia.org/wiki/Malus_domestica")
wt_wikipedia_parse(pg)
wt_wikipedia_parse(pg, tidy = TRUE)

# search wikipedia
# FIXME: utf=FALSE for now until curl::curl_escape fix
# https://github.com/jeroen/curl/issues/228
wt_wikipedia_search(query = "Pinus", utf8=FALSE)
wt_wikipedia_search(query = "Pinus", wiki = "fr", utf8=FALSE)
wt_wikipedia_search(query = "Pinus", wiki = "br", utf8=FALSE)

## curl options
# wt_wikipedia_search(query = "Pinus", verbose = TRUE, utf8=FALSE)

## use search results to dig into pages
res <- wt_wikipedia_search(query = "Pinus", utf8=FALSE)
lapply(res$query$search$title[1:3], wt_wikipedia)

## End(Not run)
```

wt_wikispecies WikiSpecies

Description

WikiSpecies

Usage

```
wt_wikispecies(name, utf8 = TRUE, ...)

wt_wikispecies_parse(
  page,
  types = c("langlinks", "iwlinks", "externallinks", "common_names", "classification"),
  tidy = FALSE
)

wt_wikispecies_search(query, limit = 10, offset = 0, utf8 = TRUE, ...)
```

Arguments

name	(character) Wiki name - as a page title, must be length 1
utf8	(logical) If TRUE, encodes most (but not all) non-ASCII characters as UTF-8 instead of replacing them with hexadecimal escape sequences. Default: TRUE
...	curl options, passed on to httr::GET()
page	(httr::response()) Result of wt_wiki_page()
types	(character) List of properties to parse
tidy	(logical). tidy output to data.frame's if possible. Default: FALSE
query	(character) query terms
limit	(integer) number of results to return. Default: 10
offset	(integer) record to start at. Default: 0

Value

wt_wikispecies returns a list, with slots:

- langlinks - language page links
- externallinks - external links
- common_names - a data.frame with name and language columns
- classification - a data.frame with rank and name columns

wt_wikispecies_parse returns a list

wt_wikispecies_search returns a list with slots for continue and query, where query holds the results, with query$search slot with the search results

References

https://www.mediawiki.org/wiki/API:Search for help on search

Examples

```
## Not run:
# high level
wt_wikispecies(name = "Malus domestica")
wt_wikispecies(name = "Pinus contorta")
wt_wikispecies(name = "Ursus americanus")
wt_wikispecies(name = "Balaenoptera musculus")

# low level
pg <- wt_wiki_page("https://species.wikimedia.org/wiki/Abelmoschus")
wt_wikispecies_parse(pg)

# search wikispecies
# FIXME: utf=FALSE for now until curl::curl_escape fix
# https://github.com/jeroen/curl/issues/228
wt_wikispecies_search(query = "pine tree", utf8=FALSE)

## use search results to dig into pages
res <- wt_wikispecies_search(query = "pine tree", utf8=FALSE)
lapply(res$query$search$title[1:3], wt_wikispecies)

## End(Not run)
```

wt_wiki_page *Get MediaWiki Page from API*

Description

Supports both static page urls and their equivalent API calls.

Usage

```
wt_wiki_page(url, ...)
```

Arguments

url	(character) MediaWiki page url.
...	Arguments passed to wt_wiki_url_build() if url is a static page url.

Details

If the URL given is for a human readable html page, we convert it to equivalent API call - if URL is already an API call, we just use that.

Value

an `HttpResponse` response object from **crul**

See Also

Other MediaWiki functions: `wt_wiki_page_parse()`, `wt_wiki_url_build()`, `wt_wiki_url_parse()`

Examples

```
## Not run:
wt_wiki_page("https://en.wikipedia.org/wiki/Malus_domestica")

## End(Not run)
```

`wt_wiki_page_parse` *Parse MediaWiki Page*

Description

Parses common properties from the result of a MediaWiki API page call.

Usage

```
wt_wiki_page_parse(
  page,
  types = c("langlinks", "iwlinks", "externallinks"),
  tidy = FALSE
)
```

Arguments

`page`	(crul::HttpResponse) Result of `wt_wiki_page()`
`types`	(character) List of properties to parse.
`tidy`	(logical). tidy output to data.frames when possible. Default: `FALSE`

Details

Available properties currently not parsed: title, displaytitle, pageid, revid, redirects, text, categories, links, templates, images, sections, properties, ...

Value

a list

See Also

Other MediaWiki functions: `wt_wiki_page()`, `wt_wiki_url_build()`, `wt_wiki_url_parse()`

Examples

```
## Not run:
pg <- wt_wiki_page("https://en.wikipedia.org/wiki/Malus_domestica")
wt_wiki_page_parse(pg)

## End(Not run)
```

wt_wiki_url_build *Build MediaWiki Page URL*

Description

Builds a MediaWiki page url from its component parts (wiki name, wiki type, and page title). Supports both static page urls and their equivalent API calls.

Usage

```
wt_wiki_url_build(
  wiki,
  type = NULL,
  page = NULL,
  api = FALSE,
  action = "parse",
  redirects = TRUE,
  format = "json",
  utf8 = TRUE,
  prop = c("text", "langlinks", "categories", "links", "templates", "images",
   "externallinks", "sections", "revid", "displaytitle", "iwlinks", "properties")
)
```

Arguments

`wiki`	(character | list) Either the wiki name or a list with `$wiki`, `$type`, and `$page` (the output of `wt_wiki_url_parse()`).
`type`	(character) Wiki type.
`page`	(character) Wiki page title.
`api`	(boolean) Whether to return an API call or a static page url (default). If `FALSE`, all following (API-only) arguments are ignored.
`action`	(character) See `https://en.wikipedia.org/w/api.php` for supported actions. This function currently only supports "parse".
`redirects`	(boolean) If the requested page is set to a redirect, resolve it.
`format`	(character) See `https://en.wikipedia.org/w/api.php` for supported output formats.
`utf8`	(boolean) If `TRUE`, encodes most (but not all) non-ASCII characters as UTF-8 instead of replacing them with hexadecimal escape sequences.

prop (character) Properties to retrieve, either as a character vector or pipe-delimited string. See https://en.wikipedia.org/w/api.php?action=help&modules=parse for supported properties.

Value

a URL (character)

See Also

Other MediaWiki functions: wt_wiki_page_parse(), wt_wiki_page(), wt_wiki_url_parse()

Examples

```
wt_wiki_url_build(wiki = "en", type = "wikipedia", page = "Malus domestica")
wt_wiki_url_build(
  wt_wiki_url_parse("https://en.wikipedia.org/wiki/Malus_domestica"))
wt_wiki_url_build("en", "wikipedia", "Malus domestica", api = TRUE)
```

---

wt_wiki_url_parse *Parse MediaWiki Page URL*

Description

Parse a MediaWiki page url into its component parts (wiki name, wiki type, and page title). Supports both static page urls and their equivalent API calls.

Usage

```
wt_wiki_url_parse(url)
```

Arguments

url (character) MediaWiki page url.

Value

a list with elements:

- wiki - wiki language
- type - wikipedia type
- page - page name

See Also

Other MediaWiki functions: wt_wiki_page_parse(), wt_wiki_page(), wt_wiki_url_build()

Examples

```
wt_wiki_url_parse(url="https://en.wikipedia.org/wiki/Malus_domestica")
wt_wiki_url_parse("https://en.wikipedia.org/w/api.php?page=Malus_domestica")
```

Index