

Package ‘geoparser’

May 16, 2017

Type Package

Title Interface to the Geoparser.io API for Identifying and Disambiguating Places Mentioned in Text

Version 0.1.1

Description A wrapper for the Geoparser.io API version 0.4.0 (see <<https://geoparser.io/>>), which is a web service that identifies places mentioned in text, disambiguates those places, and returns detailed data about the places found in the text. Basic, limited API access is free with paid plans to accommodate larger workloads.

License GPL (>= 2)

LazyData TRUE

URL <http://github.com/ropensci/geoparser>

BugReports <http://github.com/ropensci/geoparser/issues>

Encoding UTF-8

RoxygenNote 6.0.1

Suggests testthat, knitr, rmarkdown

Imports dplyr, httr, jsonlite, lazyeval, tidyr, utils, purrr, stringr, digest

VignetteBuilder knitr

NeedsCompilation no

Author Maëlle Salmon [aut, cre],
Bob Rudis [ctb] (Bob Rudis reviewed the package for rOpenSci, see <https://github.com/ropensci/onboarding/issues/43>)

Maintainer Maëlle Salmon <maelle.salmon@yahoo.se>

Repository CRAN

Date/Publication 2017-05-16 14:41:38 UTC

R topics documented:

geoparser_q 2

Index**4**

geoparser_q	<i>Issue a geoparser query</i>
-------------	--------------------------------

Description

The function calls the `geoparser.io` API which identifies places mentioned in the input text (in English), disambiguates those places, and returns data about the places found in the text.

Usage

```
geoparser_q(text_input, key = geoparser_key())
```

Arguments

<code>text_input</code>	A vector of strings, where each size must be smaller than 8 KB. See <code>nchar(text_input, type = "bytes")</code> .
<code>key</code>	Your <code>geoparser.io</code> key.

Details

To get an API key, you need to register at <https://geoparser.io/pricing.html>. With an hobbyist account, you can make up to 1,000 calls a month to the API. For ease of use, save your API key as an environment variable as described at https://stat545-ubc.github.io/bit003_api-key-env-var.html.

The package will conveniently look for your API key using `Sys.getenv("GEOPARSER_KEY")` so if your API key is an environment variable called "GEOPARSER_KEY" you don't need to input it manually.

`Geoparser.io` works best on complete sentences in English. If you have a very short text, such as a partial address like "Auckland New Zealand," you probably want to use a geocoder tool instead of a `geoparser`. In R, you can use the [opencage package](#) for geocoding!

Value

A list of 2 `data.frames` (`dplyr::tbl_dfs`). The first `data.frame` is called `properties` and contains:

- `apiVersion` the version of the API currently in use
- `source` the source of authority for the results
- `id` the unique id of the query
- `text_md5` MD5 hash of the text that was sent to the API.

The second `data.frame` contains the results and is called `results`. If there are no results it is totally empty, otherwise:

- `country` [ISO-3166 2-letter country code](#) for the country in which this place is located or NULL for features outside any sovereign territory.

- confidence A confidence score produced by the place name disambiguation algorithm. *Currently returns a placeholder value; subject to change.*
- name Best name for the specified location, with a preference for official/short name forms (e.g., "New York" over "NYC," and "California" over "State of California"), which may be different from exactly what appears in the text.
- admin1 A code representing the state/province-level administrative division containing this place. (From GeoNames.org: *"Most admin1 are FIPS codes. ISO codes are used for US, CH, BE and ME. UK and Greece are using an additional level between country and FIPS code. The code '00' stands for general features where no specific admin1 code is defined."*)
- type A text description of the geographic feature type — see <http://www.geonames.org/> for a complete list. *Subject to change.*
- geometry.type Type of the geographical feature, e.g. "Point".
- longitude Longitude.
- latitude Latitude.
- reference1 Start (index of the first character in the place reference) – *each reference to this place name found in the input text is on one distinct line.*
- reference2 End (index of the first character after the place reference) – *each reference to this place name found in the input text is on one distinct line.*
- text_md5 MD5 hash of the text that was sent to the API.

References

[Geoparser.io documentation](#)

Examples

```
## Not run:
geoparser_q(text_input = "I was born in Vannes but I live in Barcelona.")
geoparser_q(text_input = c("I was born in Vannes but I live in Barcelona.",
"France is the most beautiful place in the world.))

## End(Not run)
```

Index

geoparser_q, [2](#)