Package: refsplitr (via r-universe)

October 15, 2026

Type Package

Title author name disambiguation, author georeferencing, and mapping of coauthorship networks with 'Web of Science' data

Version 1.2.3

Description Tools to parse and organize reference records downloaded from the 'Web of Science' citation database into an R-friendly format, disambiguate the names of authors, geocode their locations, and generate/visualize coauthorship networks. This package has been peer-reviewed by rOpenSci (v. 1.0).

License GPL (>= 3)

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

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

Depends R (>= 2.10)

Imports dplyr, ggmap, ggplot2, Hmisc, igraph, Matrix, network, stringdist, rworldmap, sna, tidygeocoder

Suggests covr, gdtools, knitr, mapproj, rmarkdown, testthat, utils

VignetteBuilder knitr

Remotes dkahle/ggmap

Encoding UTF-8

RoxygenNote 7.3.2

X-schema.org-keywords name disambiguation, bibliometrics, coauthorship, collaboration, georeferencing, metascience, references, scientometrics, science of science, Web of Science

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

Roxygen list(markdown = TRUE)

LazyData true

Config/pak/sysreqs cmake libgdal-dev gdal-bin libgeos-dev libglpk-dev make libicu-dev libjpeg-dev libpng-dev libuv1-dev libxml2-dev libssl-dev libproj-dev libsqlite3-dev

Repository https://ropensci.r-universe.dev

Date/Publication 2026-07-17 02:16:39 UTC

RemoteUrl https://github.com/ropensci/refsplitr

RemoteRef master

RemoteSha 975a2c0521ee505a7d041fadac8ff5ed9755c3a1

Contents

authors_clean . . . 2
authors_georef . . . 3
authors_refine . . . 4
BITR . . . 5
BITR_geocode . . . 6
countries . . . 7
plot_addresses_country . . . 8
plot_addresses_points . . . 8
plot_net_address . . . 9
plot_net_coauthor . . . 10
plot_net_country . . . 11
references_read . . . 12

Index **13**

`authors_clean` *Seperates author information in references files from* `references_read`

Description

`authors_clean` This function takes the output from `references_read` and cleans the author information.

Usage

```
authors_clean(references)
```

Arguments

`references`	output from `references_read`

Details

Information on addresses, emails, ORCIDs, etc are matched.

It then attempts to match same author entries together into likely author groups based on common full names, addresses, emails, ORCIDs etc.

Records that are not matched this way have a Jaro-Winkler similiarty analysis metric calculated for all possible matching author names.

This calculates the amount of character similarities based on distance of similar character.

Examples

```
## Load the refsplitr sample dataset "BITR"
data(BITR)
BITR_clean <- authors_clean(BITR)

## The output of authors_clean is a list with two elements,
## which can be assigend to dataframes.
BITR_review_df <- BITR_clean$review
BITR_prelim_df <- BITR_clean$prelim

## Users can save the these dataframes outside of R as .csv files.
## The "review_df.csv" is then used to review the groupID or authorID
## assignments and make any necessary corrections.
## The function "authors_refine" is used to load and merge the changes
## into R and create a dataframe used for analyses.
```

---

authors_georef *Extracts the lat and long for each address from authors_clean*

Description

`authors_georef` This function takes the final author list from refine_authors, and calculates the lat long of the city, country, and postal code (for USA addresses) or city and country (for addresses outside the USA).

Usage

```
authors_georef(data, address_column = "address", google_api = FALSE)
```

Arguments

`data`	dataframe from `authors_refine()`
`address_column`	name of column in quotes where the addresses are
`google_api`	if `google_api = FALSE` georeferencing is carried out with the `tidygeocoder` package (option `geocode()` with `method = 'osm'`). If `google_api = TRUE`, then geocoding is done with the Google Maps API. Defaults to `FALSE`.

Details

The output is a list of three data.frames `addresses` All info from 'refine_authors' plus new columns with lat & long. It includes ALL addresses, including those that could not be geocoded. `missing_addresses` A data frame of the addresses that could NOT be geocoded. `no_missing_addresses` the `addresses` data frame with ONLY the addresses that were geocoded.

Examples

```
## Not run:
BITR_georef_df <- authors_georef(BITR_refined, address_column = "address",
google_api=FALSE)

## End(Not run)
```

`authors_refine` *Refines the authors code output from authors_clean()*

Description

`authors_refine` This function takes the author list output after the output has been synthesized for incorrect author matches. It contains a similarity score cutoff like read_authors. This however is to further constrain the list. New values ARE NOT created, instead it filters by the sim_score column in the output file.

Usage

```
authors_refine(review, prelim, sim_score = NULL, confidence = NULL)
```

Arguments

`review`	the `review` element from list output by `authors_clean`
`prelim`	the `prelim` element from list output by `authors_clean`
`sim_score`	similarity score cut off point. Number from 0-1.
`confidence`	confidence score cut off point. Number from 0 - 10.

Examples

```
## First gather the authors data.frame from authors_clean
data(BITR)
BITR_authors <- authors_clean(BITR)
BITR_review_df <- BITR_authors$review
BITR_prelim_df <- BITR_authors$prelim

## If accepting the preliminary disambiguation
## from authors_clean() without review:
refine_df <- authors_refine(BITR_review_df, BITR_prelim_df,
    sim_score = 0.90, confidence = 5)
```

```
## Note that 'sim_score' and 'confidence' are optional arguments and are
## only required if changing the default values.
refine_df <- authors_refine(BITR_review_df, BITR_prelim_df)


## If changes were made to groupID or authorID in the "_review.csv" file:
## then incorporate those changes in a text editor, save the corrections as
## a new file name, load in to R and run `authors_refine()` with the
## new corrections as the review arguement.
```

BITR *Data from the journal Biotropica (pulled from Web of Knowledge)*

Description

A dataset containing 10 articles taken from the journal Biotropica. This dataset represents the typical formatted output from `references_read()` in the refsplitr package. It serves as a testbed for commonly miscategorized names

Usage

```
BITR
```

Format

A data frame with 10 rows and 32 variables:

filename the original filename the text was created from

refID the unique identifier given to each reference article by references_read()

AB Abstract

AF Full Names

AU Abbreviated names

C1 Addresses

EM emails

RI Web of Science ID

OI OrcID

RP Reprint Address

TI Title

UT Web of Knowledge Unique ID

BP See url below

CR See url below

DE See url below

DI See url below

EP See url below

FN See url below

FU See url below

PD See url below

PG See url below

PT See url below

PU See url below

PY See url below

PM See url below

SC See url below

SN See url below

SO See url below

TC See url below

VL See url below

WC See url below

Z9 See url below The remaining codes are described on the Web of Knowledge website: https://images.webofknowledge.com/images/help/WOS/hs_wos_fieldtags.html

---

`BITR_geocode` *Georeferenced data from the journal Biotropica (pulled from Web of Science)*

Description

A dataset containing 41 authors taken from the Biotropica journal. This dataset represents the typical formatted output from `authors_georef()` in the refsplitr package. It serves as a useful testing data set for spatial functions and

Usage

```
BITR_geocode
```

Format

A data frame with 41 rows and 15 variables:

authorID ID field populated in authors_clean

university also can be considered institution for non-universities

postal_code character, international postcode

country country name

lat numeric, latitude populated from authors_georef

lon numeric, longitude populated from authors_georef

groupID ID field for what name group the author is identified as from authors_clean()

author_order numeric, order of author from journal article

address address of references pulled from the original raw WOS file

department department which is nested within university

RP_address reprint address, pulled from the original raw WOS file

RI ResearcherID number, identifier given by web of science only, less common than OrcID

OI OrcID, unique identifier for researcher given by https://orcid.org

UT unique identifier to each article, given by WOS

refID unique identifier for each article, given by references_read()

`countries` *Names of all the countries in the world*

Description

#'

Usage

```
countries
```

Format

a character vector of country names

countries a character vector of country names

@export countries @noRd

plot_addresses_country

Plot addresses, the number of which are summed by country_name

Description

This function plots an addresses data.frame object by country name.

Usage

```
plot_addresses_country(data, mapRegion = "world")
```

Arguments

data	address element from the output from the authors_georef() function, containing geocoded address latitude and longitude locations.
mapRegion	what portion of the world map to show. possible values include "world", "North America", "South America", "Australia", "Africa", "Antarctica", and "Eurasia"

Examples

```
## Using the output of authors_georef (e.g., BITR_geocode)
data(BITR_geocode)
## Plots the whole world
plot_addresses_country(BITR_geocode)

## Just select North America
plot_addresses_country(BITR_geocode, mapRegion = 'North America')
```

plot_addresses_points *Plot address point locations on world map*

Description

This function plots an addresses data.frame object by point overlaid on the countries of the world.

Usage

```
plot_addresses_points(data, mapCountry = NULL)
```

Arguments

data	the address element from the list output by the ‘authors_georef()“ function, containing geocoded address latitude and longitude locations.
mapCountry	What country to map. Possible values include "USA", "Brazil", "Australia", and "UK" use data(countries) to see possible names. No value defaults to the world map.

Examples

```
## Using the output of authors_georef (e.g., BITR_geocode)
data(BITR_geocode)
## Plots the whole world
plot_addresses_points(BITR_geocode)

## mapCountry names can be querried using:
data(countries)

## Plot only Brazil
plot_addresses_points(BITR_geocode, mapCountry = 'Brazil')
```

---

plot_net_address *Creates a network diagram of coauthors' addresses linked by reference, and with nodes arranged geographically*

Description

This function takes an addresses data.frame, links it to an authors__references dataset and plots a network diagram generated for individual points of co-authorship.

Usage

```
plot_net_address(
  data,
  mapRegion = "world",
  lineResolution = 10,
  lineAlpha = 0.5
)
```

Arguments

data	the address element from the list outputted from the authors_georef() function, containing geocoded address latitude and longitude locations.
mapRegion	what portion of the world map to show. possible values include "world", "North America", "South America", "Australia", "Africa", "Antarctica", "Eurasia"
lineResolution	the resolution of the lines drawn, higher numbers will make smoother curves default is 10.
lineAlpha	transparency of the lines, fed into ggplots alpha value. Number between 0 - 1.

Examples

```
## Using the output of authors_georef (e.g., BITR_geocode)
data(BITR_geocode)
## Plots the whole world
output <- plot_net_address(BITR_geocode)

## Just select North America
output <- plot_net_address(BITR_geocode, mapRegion = 'North America')

## Change the transparency of lines by modifying the lineAlpha parameter
output <- plot_net_address(BITR_geocode, lineAlpha = 0.2)

## Change the curvature of lines by modifying the lineResolution paramater
output <- plot_net_address(BITR_geocode, lineResolution = 30 )

output <- plot_net_address(BITR_geocode, mapRegion = 'North America', lineAlpha = 0.2,
                lineResolution = 30)
```

`plot_net_coauthor` *Creates a network diagram of coauthors' countries linked by reference This function takes an addresses data.frame, links it to an authors_references dataset and plots a network diagram generated for co-authorship.*

Description

Creates a network diagram of coauthors' countries linked by reference This function takes an addresses data.frame, links it to an authors_references dataset and plots a network diagram generated for co-authorship.

Usage

```
plot_net_coauthor(data)
```

Arguments

`data`	the `address` element from the list outputted from the 'authors_georef()" function, containing geocoded address latitude and longitude locations.

Examples

```
## Using the output of authors_georef (e.g., BITR_geocode)
data(BITR_geocode)
plot_net_coauthor(BITR_geocode)
```

plot_net_country *Creates a network diagram of coauthors' countries linked by reference, #and with nodes arranged geographically*

Description

This function takes an addresses data.frame, links it to an authors_references dataset and plots a network diagram generated for countries of co-authorship.

Usage

```
plot_net_country(
  data,
  lineResolution = 10,
  mapRegion = "world",
  lineAlpha = 0.5
)
```

Arguments

data	the address element from the list outputted from the authors_georef() function, containing geocoded address latitude and longitude locations.
lineResolution	the resolution of the lines drawn, higher numbers will make smoother curves default is 10.
mapRegion	what portion of the world map to show. possible values include "world", "North America", "South America", "Australia", "Africa", "Antarctica", and "Eurasia"
lineAlpha	transparency of the lines, fed into ggplots alpha value. Number between 0 - 1.

Examples

```
## Using the output of authors_georef (e.g., BITR_geocode)
data(BITR_geocode)
## Plots the whole world
output <- plot_net_country(BITR_geocode)

## Mapping only North America
output <- plot_net_country(BITR_geocode, mapRegion = 'North America')

## Change the transparency of lines by modifying the lineAlpha parameter
output <- plot_net_country(BITR_geocode, lineAlpha = 0.2)

## Change the curvature of lines by modifying the lineResolution paramater
output <- plot_net_country(BITR_geocode, lineResolution = 30 )

## With all arguments:
output <- plot_net_country(BITR_geocode, mapRegion = 'North America', lineAlpha = 0.2,
                lineResolution = 30)
```

`references_read` *Reads Thomson Reuters Web of Knowledge/Science and ISI reference export files (both .txt or .ciw format accepted)*

Description

`references_read` This function reads Thomson Reuters Web of Knowledge and ISI format reference data files into an R-friendly data format. The resulting dataframe is the argument for the refsplitr function `authors_clean()`.

Usage

```
references_read(data = ".", dir = FALSE, include_all = FALSE)
```

Arguments

`data`	the location of the file or files to be imported. This can be either the absolute or relative name of the file (for a single file) or folder (for multiple files stored in the same folder; used in conjunction with 'dir = TRUE"). If left blank it is assumed the location is the working directory.
`dir`	if FALSE it is assumed a single file is to be imported. Set to TRUE if importing multiple files (the path to the folder in which files are stored is set with 'data="; all files in the folder will be imported). Defaults to FALSE.
`include_all`	if FALSE only a subset of commonly used fields from references records are imported. If TRUE then all fields from the reference records are imported. Defaults to FALSE. The additional data fields included if `include_all=TRUE`: CC, CH, CL, CT, CY, FX, GA, J9, LA, PA, PI, PN, PS, RID, SU, VR, OA.

Examples

```
## If a single files is being imported from a folder called "data" located in an RStudio Project:
## imported_refs<-references_read(data = './data/refs.txt', dir = FALSE, include_all=FALSE)

## If multiple files are being imported from a folder named "heliconia" nested within a folder
## called "data" located in an RStudio Project:
## heliconia_refs<-references_read(data = './data/heliconia', dir = TRUE, include_all=FALSE)

## To load the Web of Science records used in the examples in the documentation
BITR_data_example <- system.file("extdata", "BITR_test.txt", package = "refsplitr")
BITR <- references_read(BITR_data_example)
```

Index

∗ **datasets**
BITR, 5
BITR_geocode, 6
countries, 7

authors_clean, 2
authors_georef, 3
authors_refine, 4

BITR, 5
BITR_geocode, 6

countries, 7

plot_addresses_country, 8
plot_addresses_points, 8
plot_net_address, 9
plot_net_coauthor, 10
plot_net_country, 11

references_read, 12