Package 'GDELTtools'

January 20, 2025

Type Package

Title Download, Slice, and Normalize GDELT V1 Event and Sentiment API
Data

Version 1.7

Date 2023-09-28

Author Stephen R. Haptonstahl, Thomas Scherer, Timo Thoms, and Patrick Wheatley

Maintainer Stephen R. Haptonstahl <srh@haptonstahl.org>

Description The GDELT V1 Event data set is over 41 GB now and growing 250 MB
a month. The number of source articles has increased over time and unevenly
across countries. This package makes it easy to download a subset of that
data, then normalize that data to facilitate valid time series analysis.

License MIT + file LICENSE

Depends R (>= 4.0), utils

Imports plyr, dplyr, datetimeutils, stringr

Suggests testthat

RoxygenNote 7.2.3

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2023-09-29 07:40:02 UTC

Contents

GetAllOfGDELT	*Download all the GDELT V1 Event files to a local folder*

Description

Downloads all GDELT V1 Event files not already present locally. ** This takes a long time and a lot of space. **

Usage

```
GetAllOfGDELT(
  local_folder,
  data_url_root = "http://data.gdeltproject.org/events/",
  force = FALSE
)
```

Arguments

`local_folder`	character, path to the file to be validated.
`data_url_root`	character, URL for the folder with GDELT data files.
`force`	logical, if TRUE then the download is carried out without further prompting the user.

Value

logical, TRUE if all files were downloaded successfully.

Author(s)

Stephen R. Haptonstahl   <srh@haptonstahl.org>

References

GDELT: Global Data on Events, Location and Tone, 1979-2013. Presented at the 2013 meeting of the International Studies Association in San Francisco, CA. https://www.gdeltproject.org/

Examples

```
## Not run:
GetAllOfGDELT("~/gdeltdata")
## End(Not run)
```

GetGDELT *Download and subset GDELT V1 event data*

Description

Download the GDELT V1 Event files necessary for a data set, import them, filter on various criteria, and return a data.frame.

Usage

```
GetGDELT(
  start_date,
  end_date = start_date,
  row_filter,
  ...,
  local_folder = tempdir(),
  max_local_mb = Inf,
  data_url_root = "http://data.gdeltproject.org/events/",
  verbose = TRUE
)
```

Arguments

start_date	character, earliest date to include in "YYYY-MM-DD" format.
end_date	character, latest date to include in "YYYY-MM-DD" format.
row_filter	<data-masking> Row selection. Expressions that return a logical value, and are defined in terms of the variables in GDELT. If multiple expressions are included, they are combined with the & operator. Only rows for which all conditions evaluate to TRUE are kept.
...	<tidy-select>, Column selection. This takes the form of one or more unquoted expressions separated by commas. Variable names can be used as if they were positions in the data frame, so expressions like x:y can be used to select a range of variables.
local_folder	character, if specified, where downloaded files will be saved.
max_local_mb	numeric, the maximum size in MB of the downloaded files that will be retained.
data_url_root	character, URL for the folder with GDELT data files.
verbose	logical, if TRUE then indications of progress will be displayed_

Details

Dates are parsed with guess_datetime in the datetimeutils package. The recommended format is "YYYY-MM-DD".

If local_folder is not specified then downloaded files are stored in tempdir(). If a needed file has already been downloaded to local_folder then this file is used instead of being downloaded. This can greatly speed up future downloads.

Value

data.frame

Filtering Results

The `row_filter` is passed to [filter](). This is a very flexible way to filter the rows. It's well worth checking out the [filter]() documentation.

Selecting Columns

The `...` is passed to [select](). This is a very flexible way to choose which columns to return. It's well worth checking out the [select]() documentation.

Author(s)

Stephen R. Haptonstahl	<srh@haptonstahl.org>
Thomas Scherer	<tscherer@princeton.edu>
John Beieler	<jub270@psu.edu>

References

GDELT: Global Data on Events, Location and Tone, 1979-2013. Presented at the 2013 meeting of the International Studies Association in San Francisco, CA. https://www.gdeltproject.org/

Examples

```
## Not run:
df1 <- GetGDELT(start_date="1979-01-01", end_date="1979-12-31")

df2 <- GetGDELT(start_date="1979-01-01", end_date="1979-12-31",
                row_filter=ActionGeo_CountryCode=="US")

df3 <- GetGDELT(start_date="1979-01-01", end_date="1979-12-31",
          row_filter=Actor2Geo_CountryCode=="RS" & NumArticles==2 & is.na(Actor1CountryCode),
                1:5)

df4 <- GetGDELT(start_date="1979-01-01", end_date="1979-12-31",
                row_filter=Actor2Code=="COP" | Actor2Code=="MED",
                contains("date"), starts_with("actor"))

# Specify a local folder to store the downloaded files
df5 <- GetGDELT(start_date="1979-01-01", end_date="1979-12-31",
                row_filter=ActionGeo_CountryCode=="US",
                local_folder = "~/gdeltdata")

## End(Not run)
```

GetGDELTStability	*Download data from the GDELT Stability Dashboard API to memory*

Description

Download data from the GDELT Stability Dashboard API to memory

Usage

```
GetGDELTStability(
  location,
  var_to_get = c("instabiliity", "conflict", "protest", "tone", "artvolnorm"),
  time_resolution = c("day", "15min"),
  smoothing = 1,
  num_days = ifelse(time_resolution == "day", 180, 7),
  multi_ADM1 = FALSE
)
```

Arguments

location	character, two-digit country code or four-digit ADM1 code (see below).
var_to_get	character, variable to download (see below).
time_resolution	
	character, either "day" or "15min".
smoothing	numeric, integer number of time_resolution periods to smooth over.
num_days	numeric, number of days of data to download.
multi_ADM1	logical, if TRUE then var_to_get will be downloaded for all ADM1 codes in the country (specified in location).

Value

data.frame

location

This is a single location code, either from http://data.gdeltproject.org/blog/stability-dashboard-api/GEOLOOKUP-COUNTRY.TXT or http://data.gdeltproject.org/blog/stability-dashboard-api/GEOLOOKUP-ADM1.TXT

var_to_get

One of:

- "instability": This display a simple synthetic "instability" measure for a country offering a very basic, but insightful, view of the current level of conflict and instability involving it. Currently it is calculated by summing the total number of QuadClass=MaterialConflict and EventRootCode=14(Protest)

events together and dividing by the total number of all events worldwide monitored by GDELT in the same time period. This yields a normalized view of instability.

- "conflict": Same as above, but only includes QuadClass=MaterialConflict, ignoring protest events.

- "protest": Same as above, but only includes EventRootCode=14, assessing only protest activity, but excluding all other kinds of conflict.

- "tone": Average Standard GDELT Tone of all articles mentioning the location at least twice in the article within the given timeframe. This uses a very basic filter of requiring that an article mention the location at least twice anywhere in the article body, and assesses tone at the article level. Currently only the Standard GDELT Tone emotion is available, but in the future we hope to integrate the entire array of GCAM emotions. This variable can be especially insightful to spotting deteriorating situations where coverage of a country or area is turning increasingly negative, even if physical unrest has ceased or not yet begun.

- "artvolnorm": This tallies the total number of articles mentioning the location at least twice anywhere in the article, divided by the total number of articles monitored by GDELT in the given timeframe, offering a normalized view of attention being paid to the location regardless of any physical unrest or other activity occurring there. This variable offers a useful measure of changes in overall global "attention" being paid to a given location.

Author(s)

Stephen R. Haptonstahl   <srh@haptonstahl.org>

References

GDELT Stability Dashboard API <https://blog.gdeltproject.org/announcing-the-gdelt-stability-dashboard-a>

Examples

```
## Not run:
ex1 <- GetGDELTStability(location="FR",
                         var_to_get="tone",
                         time_resolution="day",
                         smoothing=1,
                         num_days=10)

ex2 <- GetGDELTStability(location="IS",
                         var_to_get="protest",
                         time_resolution="15min",
                         smoothing=3,
                         num_days=1)

ex3 <- GetGDELTStability(location="AR",
                         var_to_get="conflict",
                         time_resolution="day",
                         smoothing=1,
```

```
                            num_days=10,
                            multi_ADM1=TRUE)
## End(Not run)
```

NormEventCounts	*Scale event counts*

Description

Scale event counts based on the unit of analysis.

Usage

```
NormEventCounts(x, unit_analysis, var_name = "norming_vars")
```

Arguments

x	data.frame, a GDELT data.frame.
unit_analysis	character, default is country_day; other options: country_month, country_year, day, month, year
var_name	character, base name for the new count variables

Details

For `unit_analysis`, day and country-day put out a data set where date is of class 'date'. All other options put out a data set where year or month is integer (this needs to be unified in a later version).

Value

data.frame

Author(s)

Oskar N.T. Thoms	<othoms@princeton.edu>
Stephen R. Haptonstahl	<srh@haptonstahl.org>
John Beieler	<jub270@psu.edu>

References

GDELT: Global Data on Events, Location and Tone, 1979-2012. Presented at the 2013 meeting of the International Studies Association in San Francisco, CA. <https://www.gdeltproject.org/>

Examples

```
## Not run:
GDELT_subset_data <- GetGDELT("2013-06-01", "2013-06-07",
 (ActionGeo_CountryCode=="AF" | ActionGeo_CountryCode=="US") & EventCode>=140 & EventCode<150,
  local_folder="~/gdeltdata")
GDELT_normed_data <- NormEventCounts(x = GDELT_subset_data,
  unit_analysis="day",
  var_name="protest")
## End(Not run)
```

Index