Package 'graphTweets'

September 28, 2018

Type Package

Title Visualise Twitter Interactions

Version 0.5.0

Date 2018-08-25

Description Allows building an edge table from data frame of tweets,
also provides function to build nodes and another create a temporal graph.

License MIT + file LICENSE

Depends R (>= 3.2.0)

Imports dplyr, igraph, rtweet, purrr, magrittr, utils, tidyr, zeallot,
combinat

RoxygenNote 6.1.0

URL <http://graphTweets.john-coene.com>

BugReports <https://github.com/JohnCoene/graphTweets/issues>

Suggests testthat, htmltools

Encoding UTF-8

NeedsCompilation no

Author John Coene [aut, cre]

Maintainer John Coene <jcoenep@gmail.com>

Repository CRAN

Date/Publication 2018-08-25 17:02:18 UTC

R topics documented:

---

gt_collect   *Collect*

Description

Collect

Usage

```
gt_collect(gt)
```

Arguments

gt   An object of class graphTweets as returned by gt_edges and gt_nodes.

Value

A named list of tibble 1) edges and 2) nodes.

Examples

```
# simulate dataset
tweets <- data.frame(
  text = c("I tweet @you about @him",
           "I tweet @me about @you"),
  screen_name = c("me", "him"),
  retweet_count = c(19, 5),
  status_id = c(1, 2),
  stringsAsFactors = FALSE
)

tweets %>%
  gt_edges(text, screen_name, status_id) %>%
  gt_nodes() %>%
  gt_collect() -> net
```

---

gt_dyn   *Dynamise*

Description

Create a dynamic graph to import in Gephi.

Usage

```
gt_dyn(gt, lifetime = Inf)
```

Arguments

gt   An object of class graphTweets as returned by gt_edges and gt_nodes.
lifetime   Lifetime of a tweet in milliseconds, defaults to Inf.

Examples

```
## Not run:
# simulate dataset
tweets <- data.frame(
  text = c("I tweet @you about @him and @her",
           "I tweet @me about @you"),
  screen_name = c("me", "him"),
  created_at = c(Sys.time(), Sys.time() + 10000),
  status_id = c(1, 2),
  stringsAsFactors = FALSE
)

tweets %>%
  gt_edges(text, screen_name, status_id, "created_at") %>%
  gt_nodes() %>%
  gt_dyn() %>%
  gt_collect() -> net

## End(Not run)
```

---

gt_edges	*Edges*

Description

Get edges from data.frame of tweets.

Usage

```
gt_edges(data, source, target, ..., tl = TRUE)

gt_edges_bind(gt, source, target, ..., tl = TRUE)

gt_co_edges(data, col, tl = TRUE)

gt_co_edges_bind(gt, col, tl = TRUE)
```

Arguments

data	Data.frame of tweets, usually returned by the rtweet package.
source	Author of tweets.
target	Edges target.
...	any other column name, see examples.
tl	Set to TRUE to convert hashtags to lower case.
gt	An object of class graphTweets as returned by gt_edges and gt_nodes.
col	Column containing co-mentions.

---

gt_edges_   *Deprecated Functions*

Description

These functions are deprecated, see gt_edges and gt_co_edges.

Usage

```
gt_edges_(data, tweets = "text", source = "screen_name",
  id = "status_id", ...)

gt_edges_hashes(data, hashtags, tl = TRUE)

gt_edges_hashes_(data, hashtags = "hashtags", tl = TRUE)
```

Arguments

data	Data.frame of tweets, usually returned by the rtweet package.
tweets	Column containing tweets.
source	Author of tweets.
id	Unique id.
...	any other column name, see examples.
hashtags	Column containing co-mentions.
tl	Set to TRUE to convert hashtags to lower case.

---

gt_edges_from_text   *Edges from text*

Description

Get edges from data.frame of tweets.

Usage

```
gt_edges_from_text(data, id, source, tweets, ...)

gt_edges_from_text_(data, id = "status_id", source = "screen_name",
  tweets = "text", ...)
```

Arguments

data	Data.frame of tweets, usually returned by the rtweet package.
id	tweets unique id.
source	Author of tweets.
tweets	Column containing tweets.
...	any other column name.

Details

The `tl` arguments stands for [tolower](link) and allows converting the #hashtags to lower case as these often duplicated, i.e.: #python #Python.

Value

An object of class `graphTweets`.

Functions

- `gt_edges` - Build networks of users.

- `gt_co_edges` - Build networks of users to hashtags.

Examples

```
# simulate dataset
tweets <- data.frame(
  text = c("I tweet @you about @him and @her",
           "I tweet @me about @you"),
  screen_name = c("me", "him"),
  retweet_count = c(19, 5),
  status_id = c(1, 2),
  hashtags = c("rstats", "Python"),
  stringsAsFactors = FALSE
)

tweets %>%
  gt_edges_from_text(status_id, screen_name, text)
```

---

gt_graph	*Graph*

Description

Build `igraph` object.

Usage

```
gt_graph(gt)
```

Arguments

gt   An object of class `graphTweets` as returned by [gt_edges](link) and [gt_nodes](link).

Value

An object of class `igraph`.

Examples

```
# simulate dataset
tweets <- data.frame(
  text = c("I tweet @you about @him",
           "I tweet @me about @you"),
  screen_name = c("me", "him"),
  retweet_count = c(19, 5),
  status_id = c(1, 2),
  stringsAsFactors = FALSE
)

tweets %>%
  gt_edges(text, screen_name, status_id) %>%
  gt_nodes() %>%
  gt_graph() -> net
```

gt_nodes   *Nodes*

Description

Get nodes from a graphTweets object.

Usage

```
gt_nodes(gt, meta = FALSE)
```

Arguments

gt   An object of class graphTweets as returned by gt_edges and gt_nodes.

meta   Set to TRUE to add meta data to nodes.

Value

An object of class graphTweets.

gt_save   *Save*

Description

Save the graph to file.

Usage

```
gt_save(gt, file = "graphTweets.graphml", format = "graphml", ...)
```

Arguments

gt	An object of class graphTweets as returned by [gt_edges](#) and [gt_nodes](#).
file	File name including extension (format).
format	Format file format, see [write_graph](#).
...	Any other argument to pass to [write_graph](#).

Examples

```
## Not run:
# simulate dataset
tweets <- data.frame(
  text = c("I tweet @you about @him",
           "I tweet @me about @you"),
  screen_name = c("me", "him"),
  retweet_count = c(19, 5),
  created_at = c(Sys.time(), Sys.time() + 15000),
  status_id = c(1, 2),
  stringsAsFactors = FALSE
)

tweets %>%
  gt_edges(text, screen_name, "created_at") %>%
  gt_nodes(TRUE) %>%
  gt_dyn() %>%
  gt_save()

## End(Not run)
```

Index