Package ‘VOSONDash’

July 27, 2020

Version 0.5.7
Title User Interface for Collecting and Analysing Social Networks
Description A 'Shiny' application for the interactive visualisation and analysis of networks that also provides a web interface for collecting social media data using 'vosonSML'.
Type Package
Imports data.table, graphics, httpuv, httr, igraph (>= 1.2.2), lattice, magrittr, RColorBrewer, shiny (>= 1.3.2), SnowballC, systemfonts, syuzhet, textutils, tm, utils, vosonSML (>= 0.29.0), wordcloud
Suggests dplyr, DT, htmlwidgets, rtweet (>= 0.6.8), shinydashboard, shinyjs, visNetwork
Depends R (>= 3.2.0)
Encoding UTF-8
Author Bryan Gertzel, Robert Ackland
Maintainer Bryan Gertzel <bryan.gertzel@anu.edu.au>
License GPL (>= 3)
RoxygenNote 7.1.1
NeedsCompilation no
URL https://github.com/vosonlab/VOSONDash
BugReports https://github.com/vosonlab/VOSONDash/issues
Repository CRAN
Date/Publication 2020-07-27 13:20:02 UTC

R topics documented:

VOSONDash-package ........................................... 2
addAdditionalMeasures ..................................... 2
applyCategoricalFilters ..................................... 3
applyComponentFilter ....................................... 4
**addAdditionalMeasures**

Add additional measures to graph as vertex attributes.

**Description**

Adds degree, in-degree, out-degree, betweenness and closeness measures to graph as vertex attributes.

**Usage**

```r
addAdditionalMeasures(g)
```
applyCategoricalFilters

Arguments

- **g**: igraph graph object.

Value

An igraph graph object.

---

**Filter out graph vertices not in selected category**

Description

This function removes vertices that are not in the selected categories values list or sub-categories.

Usage

```r
applyCategoricalFilters(
  g,  
  selected_cat, 
  selected_subcats, 
  cat_prefix = "vosonCA_"
)
```

Arguments

- **g**: igraph graph object.
- **selected_cat**: Character string. Selected vertex category without prefix.
- **selected_subcats**: List. Selected sub-category values to include in graph.
- **cat_prefix**: Character string. Category attribute prefix format to match. Default is "vosonCA_".

Value

An igraph graph object.

Examples

```r
## Not run:
# return a graph containing only vertices that have the vertex category
# attribute "vosonCA_Stance" value "liberal"
g <- loadPackageGraph("DividedTheyBlog_40Alist_release.graphml")

g <- applyCategoricalFilters(g, "Stance", c("liberal"))
## End(Not run)
```
applyComponentFilter  Filter out graph vertices not in component size range

Description
This function removes any graph vertices that are in components that fall outside of the specified component size range.

Usage
applyComponentFilter(g, component_type = "strong", component_range)

Arguments
- **g**: igraph graph object.
- **component_type**: Character string. Use strongly or weakly connected components by specifying "strong" or "weak". Ignored for undirected graphs. Default is "strong".
- **component_range**: Numeric vector. Min and max values or size range of component.

Value
An igraph graph object.

applyGraphFilters  Filter out graph vertices and edges from graph object that are isolates, multi edge or edge loops

Description
This function removes isolate vertices, multiple edges between vertices and or vertex edge loops from a graph.

Usage
applyGraphFilters(g, isolates = TRUE, multi_edge = TRUE, loops_edge = TRUE)

Arguments
- **g**: igraph graph object.
- **isolates**: Logical. Include isolate vertices in graph. Default is TRUE.
- **multi_edge**: Logical. Include multiple edges between vertices in graph. Default is TRUE.
- **loops_edge**: Logical. Include vertex edge loops in graph. Default is TRUE.
**applyPruneFilter**

**Value**

An igraph graph object.

**Note**

Removing multiple edges or edge loops from a graph will simplify it and remove other edge attributes.

**Description**

This function removes a list of vertices from the graph object by vertex id value.

**Usage**

applyPruneFilter(g, selected_prune_verts)

**Arguments**

- **g**
  - **igraph** graph object.
- **selected_prune_verts**
  - List. Selected vertex ids to remove.

**Value**

An igraph graph object.

**corpusFromGraph**

Create a text corpus from graph text attribute data

**Description**

This function creates a text corpus from node or edge text attribute data in an igraph.
Usage

corpusFromGraph(
  g = NULL,
  txt_attr = NULL,
  type = "vertex",
  iconv = FALSE,
  html_decode = TRUE,
  rm_url = TRUE,
  rm_num = TRUE,
  rm_punct = TRUE,
  rm_twit_hashtags = FALSE,
  rm_twit_users = FALSE,
  sw_kind = "SMART",
  rm_words = NULL,
  stem = FALSE
)

Arguments

g an igraph graph object.
txt_attr Character string. Name of graph text attribute. Default is NULL.
type Character string. Graph attribute type. Default is "vertex".
iconv Logical. Use the iconv function to attempt UTF8 conversion. Default is FALSE.
html_decode Logical. HTML decode text. Default is TRUE.
rm_url Logical. Remove URL’s. Default is TRUE.
rm_num Logical. Remove numbers. Default is TRUE.
rm_punct Logical. Remove punctuation. Default is TRUE.
rm_twit_hashtags Logical. Remove twitter hashtags. Default is FALSE.
rm_twit_users Logical. Remove twitter user names. Default is FALSE.
sw_kind Character string. Stopword dictionary. Refer stopwords kind parameter. Default is "SMART".
rm_words Character vector. User defined stopwords. Default is NULL.
stem Logical. Apply word stemming. Default is FALSE.

Value

A tm text corpus object.
getNetworkMetrics

Description
Function creates a vector of calculated network metrics for a graph.

Usage
getNetworkMetrics(g, component_type = "strong")

Arguments

- **g**  
  igraph graph object.

- **component_type**  
  Character string. Use strongly or weakly connected components by specifying "strong" or "weak". Ignored for undirected graphs. Default is "strong".

Value
Network metrics as named vector.

getRedditUrlSubreddit

Description
This function extracts the subreddit name from a reddit thread url.

Usage
getRedditUrlSubreddit(url)

Arguments

- **url**  
  Character string. Reddit thread url.

Value
Subreddit name as character string.
**getRedditUrlThreadId**  
*Get a reddit thread id from url*

**Description**  
This function extracts the thread id from a reddit thread url.

**Usage**  
```r
getRedditUrlThreadId(url)
```

**Arguments**  
- `url`  
  Character string. Reddit thread url.

**Value**  
Reddit thread id as character string.

---

**getVertexCategories**  
*Get a list of vertex category attribute names and values*

**Description**  
This function returns a list of graph vertex attribute names that match a category attribute prefix format and their unique values.

**Usage**  
```r
gvertexCategories(g, cat_prefix = "vosonCA_")
```

**Arguments**  
- `g`  
  igraph graph object.
- `cat_prefix`  
  Character string. Category attribute prefix format to match. Default is "vosonCA_".

**Value**  
A named list of vertex category attributes and values.
getYoutubeVideoId

Examples

## Not run:
# get a list of voson vertex categories and values
g <- loadPackageGraph("DividedTheyBlog_40Alist_release.graphml")

vcats <- getVertexCategories(g)

# vcats
# $Stance
# [1] "conservative" "liberal"

## End(Not run)

getYoutubeVideoId

Get a youtube video id from url

Description

This function extracts the youtube video id from a youtube video url.

Usage

getYoutubeVideoId(url)

Arguments

url Character string. Youtube video url.

Value

Video id as character string.

loadPackageGraph

Load package included network graph

Description

This function loads a network graph included in the extdata directory of the VOSONDash package by file name.

Usage

loadPackageGraph(fname)
mixmat

Create a mixing matrix

Arguments

fname  Character string. Name of demonstration graphml file.

Value

An igraph graph object.

Examples

## Not run:
# load the "Divided They Blog" package included network graph by file name
g <- loadPackageGraph("DividedTheyBlog_40Alist_release.graphml")
## End(Not run)

Description

Function creates a mixing matrix by graph vertex attribute.

Usage

mixmat(g, attrib, use_density = TRUE)

Arguments

g  igraph graph object.
attrib  Character string. Vertex attribute or category.
use_density  Logical. Use edge density. Default is TRUE.

Value

A mixing matrix.

Note

Mixing matrix original function written by Gary Weissman. See: https://gist.github.com/gweissman/2402741.
Examples

```r
## Not run:
# create a mixing matrix of the demonstration network based on vertex
# categorical attribute for political stance "vosonCA_Stance"
# loadPackageGraph("DividedTheyBlog_40Alist_release.graphml")

mm <- mixmat(g, "vosonCA_Stance", use_density = FALSE)
## End(Not run)
```

runVOSONDash

Run the VOSON Dashboard Shiny Application

Description

This function launches the VOSONDash Shiny app in the default web browser.

Usage

```
runVOSONDash(pkgStartupMsgs = FALSE, isLocal = NULL)
```

Arguments

- `pkgStartupMsgs` Logical. Display app package loading messages. Default is `FALSE`.
- `isLocal` Logical. Manually set app local or server mode flag.

Value

None

wordCloudPlot

Create a wordcloud plot

Description

This function creates a wordcloud plot from word frequencies.

Usage

```
wordCloudPlot(
    word_freqs,
    seed = NULL,
    min_freq = 1,
    max_words = 50,
    pcolors = NULL,
    family = NULL,
    ...
)
```
wordFreqChart

Arguments

word_freqs  Table. Table of word frequencies.
seed        Numeric. Seed value can be supplied to reproduce a word cloud layout.
min_freq    Numeric. Minimum word frequency to include a word in the word cloud. Default is 1.
max_words   Numeric. Maximum number of words to render in the word cloud. Default is 50.
pcolors     List. Colors to assign categorical variable in the plot or palette to use if random.color. Default is NULL.
family      Character. Set a font family for plot labels. Default is NULL.
...

Value

A wordcloud plot.

wordFreqChart  Create a word frequency chart

Description

This function creates a horizontal barchart of word frequencies.

Usage

wordFreqChart(
  word_freqs,
  min_freq = 1,
  top_count = 20,
  pcolors = NULL,
  family = NULL
)

Arguments

word_freqs  Dataframe. Word frequencies.
min_freq    Numeric. Minimum frequency for a word to be included in the chart. Default is 1.
top_count   Numeric. Top count of words to render in word frequency chart. Default is 20.
pcolors     List. Colors to assign categorical variable in the plot. Default is NULL.
family      Character string. Set a font family for plot labels. Default is NULL.
wordFreqFromCorpus

Value

A barchart plot.

Description

Create a word frequency dataframe from a text corpus.

Usage

wordFreqFromCorpus(
  corp,
  rm_sparse = 0.99,
  word_len = c(3, 26),
  word_freq = c(1, Inf)
)

Arguments

corp a tm text corpus object.

rm_sparse Logical. Remove proportion of sparse terms. Default is 0.99.

word_len Numeric vector. Min and max length of words to include. Default is c(3, 26).

word_freq Numeric vector. Min and max frequency of words to include. Default is c(1, Inf).

Value

A data.table of word frequencies.

wordSentChart

Create an NRC emotion chart

Description

This function creates a horizontal barchart measuring and sorting the eight NRC lexicon emotions. Emotions are measured as the proportion of the total value of the eight emotions in the text as a percentage.

Usage

wordSentChart(data, pcolors = NULL)
wordSentData

Arguments

data 
  Dataframe. NRC emotions table.

pcolors 
  List. Colors to assign categorical variable in the plot. Default is NULL.

Value

A barchart plot.

Note

Uses the `syuzhet` package implementation of Saif Mohammad’s NRC Emotion lexicon.

wordSentData 

Create NRC emotion data

Description

This function creates an NRC emotion dataframe from a text corpus.

Usage

wordSentData(corp, word_len = c(3, 26))

Arguments

corp 
  tm package document Corpus object.

word_len 
  Numeric vector. Min and max length of words to include. Default is c(3, 26).

Value

An NRC sentiment dataframe.

Note

Uses the `syuzhet` package implementation of Saif Mohammad’s NRC emotion lexicon.
wordSentValenceChart

Description

This function creates a vertical barchart of the sum of negative and positive sentiments, and the valence or net sentiment in a text corpus.

Usage

wordSentValenceChart(data)

Arguments

data Dataframe. NRC emotions table.

Value

A barchart plot.
Index

addAdditionalMeasures, 2
applyCategoricalFilters, 3
applyComponentFilter, 4
applyGraphFilters, 4
applyPruneFilter, 5

Corpus, 14
corpusFromGraph, 5

getNetworkMetrics, 7
getRedditUrlSubreddit, 7
getRedditUrlThreadId, 8
getVertexCategories, 8
getYoutubeVideoId, 9

loadPackageGraph, 9

mixmat, 10

runVOSONDash, 11

VOSONDash (VOSONDash-package), 2
VOSONDash-package, 2

wordcloud::wordcloud, 12
wordCloudPlot, 11
wordFreqChart, 12
wordFreqFromCorpus, 13
wordSentChart, 13
wordSentData, 14
wordSentValenceChart, 15