Package ‘ggwordcloud’

May 30, 2024

Type Package

Title A Word Cloud Geom for 'ggplot2'

Version 0.6.2

Maintainer Erwan Le Pennec <lepennec@gmail.com>

Description Provides a word cloud text geom for 'ggplot2'. Texts are placed so that they do not overlap as in 'ggrepel'. The algorithm used is a variation around the one of 'wordcloud2.js'.

License GPL-3

Depends R (>= 3.5.0), ggplot2 (>= 3.0.0)

Imports grid, gridtext, Rcpp, scales (>= 1.0.0), colorspace, png

Suggests testthat (>= 2.0.0), knitr, rmarkdown, ggrepel, wordcloud, wordcloud2, covr, dplyr, tidyr

LinkingTo Rcpp

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

VignetteBuilder knitr

URL https://github.com/lepennec/ggwordcloud, https://lepennec.github.io/ggwordcloud/

BugReports https://github.com/lepennec/ggwordcloud/issues

NeedsCompilation yes

Author Erwan Le Pennec [aut, cre], Kamil Slowikowski [aut]

Repository CRAN

Date/Publication 2024-05-30 15:30:05 UTC

R topics documented:

geom_text_wordcloud *word cloud text geoms*

Description

geom_text_wordcloud adds text to the plot using a variation of the wordcloud2.js algorithm. The texts are layered around a spiral centred on the original position. This geom is based on geom_text_repel which in turn is based on geom_text. See the documentation for those functions for more details. By default, the font size is directly linked to the size aesthetic. geom_text_wordcloud_area is an alias, with a different set of default, that chooses a font size so that the area of the text given by the label aesthetic is linked to the size aesthetic. You can also specify a label_content aesthetic that overrides the label after its has been used to choose the font size.

Usage

```
geom_text_wordcloud(
  mapping = NULL,
  data = NULL,
  stat = "identity",
  position = "identity",
  ...,
  parse = FALSE,
  nudge_x = 0,
  nudge_y = 0,
  eccentricity = 0.65,
  rstep = 0.01,
  tstep = 0.02,
  perc_step = 0.01,
  max_steps = 10,
  grid_size = 4,
  max_grid_size = 128,
  grid_margin = 1,
  xlim = c(NA, NA),
  ylim = c(NA, NA),
  seed = NA,
  rm_outside = FALSE,
  shape = "circle",
  mask = NA,
  area_corr = FALSE,
  na.rm = FALSE,
  show.legend = FALSE,
  inherit.aes = TRUE,
  show_boxes = FALSE,
  use_richtext = TRUE
)

geom_text_wordcloud_area(
  mapping = NULL,
  data = NULL,
  stat = ”identity”,
  position = ”identity”,
  ...,
  parse = FALSE,
  nudge_x = 0,
  nudge_y = 0,
  eccentricity = 0.65,
  rstep = 0.01,
  tstep = 0.02,
  perc_step = 0.01,
  max_steps = 10,
  grid_size = 4,
  max_grid_size = 128,
  grid_margin = 1,
  xlim = c(NA, NA),
  ylim = c(NA, NA),
  seed = NA,
  rm_outside = FALSE,
  shape = ”circle”,
  mask = NA,
  area_corr = TRUE,
  na.rm = FALSE,
  show.legend = FALSE,
  inherit.aes = TRUE,
  show_boxes = FALSE,
  use_richtext = TRUE
)
```

Arguments

`mapping` Set of aesthetic mappings created by `aes` or `aes_`. If specified and `inherit.aes = TRUE` (the default), is combined with the default mapping at the top level of the plot. You only need to supply `mapping` if there isn't a mapping defined for the plot. Note that if not specified both x and y are set to 0.5, i.e. the middle of the default panel. Two non classic aesthetics are defined `angle_group` and `mask_group` which define groups used respectively to use different angular sector and different masks in the word cloud.

`data` A data frame. If specified, overrides the default data frame defined at the top level of the plot.

`stat` The statistical transformation to use on the data for this layer, as a string.

`position` Position adjustment, either as a string, or the result of a call to a position adjustment function.

`...` other arguments passed on to `layer`. There are three types of arguments you can use here:

- Aesthetics: to set an aesthetic to a fixed value, like `colour = "red"` or `size = 3`.
- Other arguments to the layer, for example you override the default `stat` associated with the layer.
- Other arguments passed on to the stat.

`parse` If `TRUE`, the labels will be parsed into expressions and displayed as described in ?plotmath

`nudge_x, nudge_y`
Horizontal and vertical adjustments to nudge the starting position of each text label.

`eccentricity` eccentricity of the spiral. Default to .65

`rstep` relative wordcloud spiral radius increment after one full rotation. Default to .01.

`tstep` wordcloud spiral angle increment at each step. Default to .02.

`perc_step` parameter used to define the minimal distance between two successive candidate positions on the ellipse. Default to .01

`max_steps` maximum number of steps avoided thanks to this minimal criterion. Default to 10. Set to 1 to recover the previous behavior

`grid_size` grid size used when creating the text bounding boxes. Default to 4

`max_grid_size` maximum size of the bounding boxes. Default to 128

`grid_margin` safety margin around the texts. Default to 1.

`xlim, ylim` Limits for the x and y axes. Text labels will be constrained to these limits. By default, text labels are constrained to the entire plot area.

`seed` Random seed passed to `set.seed`. Defaults to `NA`, which means that `set.seed` will not be called.

`rm_outside` Remove the texts that could not be fitted. Default to `FALSE`

`shape` select the shape of the clouds among `circle`, `cardioid`, `diamond`, `square`, `triangle-forward`, `triangle-upright`, `pentagon`, `star`. Default to `circle`

`mask` a mask (or a list of masks) used to define a zone in which the text should be placed. Each mask should be coercible to a raster in which non full transparency defined the text zone. When a list of masks is given, the mask_group aesthetic defines which mask is going to be used. Default to `NA`, i.e. no mask.

`area_corr` Set the font size so that the area is proportional to size aesthetic when the scale_size_area is used. As this is not the classical choice, the default is `FALSE` so that, by default, the length of the text is not taken into account. `geom_text_wordcloud_area` set this to `TRUE` by default.

`na.rm` Remove missing values if TRUE

`show.legend` is set by default to `FALSE`

`inherit.aes` Inherits aesthetics if TRUE

`show_boxes` display the bounding boxes used in the placement algorithm is set to `TRUE`. Default to `FALSE`.

`use_richtext` use the enhanced gridtext text grob instead of the grid one. Allow to use markdown/html syntax in label. Default to `TRUE`.

Value

a ggplot

Examples

```
set.seed(42)
data("love_words_latin_small")

ggplot(love_words_latin_small, aes(label = word, size = speakers)) +
geom_text_wordcloud() +
scale_size_area(max_size = 20) +
theme_minimal()

ggplot(love_words_latin_small, aes(label = word, size = speakers)) +
geom_text_wordcloud_area() +
scale_size_area(max_size = 20) +
theme_minimal()
```

`ggwordcloud` *wordcloud approximate replacement*

Description

ggwordcloud is meant as an approximate replacement for `wordcloud`. It has almost the same syntax but allows only the words/freqs input. As the underlying algorithms are not strictly equal, the resulting wordcloud is only similar to the ones one can obtain with `wordcloud`.

Usage

```
ggwordcloud(
  words,
  freq,
  scale = c(4, 0.5),
  min.freq = 3,
  max.words = Inf,
  random.order = TRUE,
  random.color = FALSE,
  rot.per = 0.1,
  colors = "black",
  ordered.colors = FALSE,
  ...
)
```

Arguments

`words`	the words
`freq`	their frequencies
`scale`	A vector of length 2 indicating the range of the size of the words.
`min.freq`	words with frequency below min.freq will not be plotted
`max.words`	Maximum number of words to be plotted. least frequent terms dropped
`random.order`	plot words in random order. If false, they will be plotted in decreasing frequency
`random.color`	choose colors randomly from the colors. If false, the color is chosen based on the frequency
`rot.per`	proportion words with 90 degree rotation
`colors`	color words from least to most frequent
`ordered.colors`	if true, then colors are assigned to words in order
`...`	Additional parameters to be passed to geom_text_wordcloud

Value

a ggplot

Examples

```
set.seed(42)
data("love_words_latin_small")

ggwordcloud(love_words_latin_small$word, love_words_latin_small$speakers)
```

`ggwordcloud2` *wordcloud2 approximate replacement*

Description

`ggwordcloud2` is meant as an approximate replacement for `wordcloud2`. It has almost the same syntax but fewer options. In particular, there is no background image (so far...). As the underlying algorithms are not strictly equal, the resulting wordcloud is only similar to the ones one can obtain with `wordcloud2`.

Usage

```
ggwordcloud2(
  data,
  size = 1,
  color = "random-dark",
  minRotation = -pi/4,
  maxRotation = pi/4,
  shuffle = TRUE,
  rotateRatio = 0.4,
  shape = "circle",
  ellipticity = 0.65,
  figPath = NA,
  ...
)
```

Arguments

`data` a dataframe whose two first columns are the names and the freqs or a table

`size` scaling factor. Default to 1

`color` color scheme either "random-dark", "random-light" or a list of color of the size of the dataframe. Default to "random-dark"

`minRotation` the minimal rotation angle

`maxRotation` the maximal rotation angle

`shuffle` if TRUE, the words are shuffled at the beginning

`rotateRatio` the proportion of rotated words

`shape` control the shape of the cloud

`ellipticity` control the eccentricity of the wordcloud

`figPath` path to an image used a mask

`...` the remaining parameters are passed to geom_text_wordcloud

Value

a ggplot

Examples

```
set.seed(42)
data("love_words_latin_small")

ggwordcloud2(love_words_latin_small[,c("word", "speakers")])
```

`love_words` *Love in several languages with number of speakers*

Description

A dataset containing the word love in different languages (147 or 34 for the small one) as well as the number of native speakers and overall speakers of those languages. Latin only version are used in the help.

Usage

```
love_words

love_words_small

love_words_latin

love_words_latin_small
```

Format

a data.frame with 147 observations (or 34 for the small one) of 5 variables

iso_639_3 the ISO 639-3 language code

word the word love in that language

name English name of the language

native_speakers number of native speakers in millions

speakers number of speakers in millions

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 34 rows and 5 columns.

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 87 rows and 5 columns.

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 14 rows and 5 columns.

Source

wikipedia

`power_trans` *A signed power transform*

Description

A signed power transform

Usage

```
power_trans(power = 1)
```

Arguments

`power`	power exponent of the direct transform

`thankyou_words` *'Thank you' in several languages with number of speakers*

Description

A dataset containing the word 'Thank you' in different languages (133 or 34 for the small one) as well as the number of native speakers and overall speakers of those languages.

Usage

```
thankyou_words

thankyou_words_small
```

Format

a data.frame with 133 observations (or 34 for the small one) of 4 variables

iso_639_3 the ISO 639-3 language code

word the word love in that language

native_speakers number of native speakers in millions

speakers number of speakers in millions

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 34 rows and 5 columns.

Source

wikipedia

Index