Package 'tidyEmoji'

August 19, 2023

Type Package

Title Discovers Emoji from Text

Version 0.1.1

Description Unicodes are not friendly to work with, and not all Unicodes are
Emoji per se, making obtaining Emoji statistics a difficult task. This
tool can help your experience of working with Emoji as smooth as possible,
as it has the 'tidyverse' style.

License GPL (>= 3)

URL https://pursuitofdatascience.github.io/tidyEmoji/

BugReports https://github.com/PursuitOfDataScience/tidyEmoji/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Depends R (>= 3.5.0)

Imports dplyr, emoji, purrr, stringr, tibble, tidyr, utils

Suggests rmarkdown, knitr, testthat (>= 3.0.0), ggplot2, readr,
forcats

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Youzhi Yu [aut, cre]

Maintainer Youzhi Yu <yuyouzhi666@icloud.com>

Repository CRAN

Date/Publication 2023-08-19 20:10:02 UTC

R **topics documented:**

category_unicode_crosswalk

Emoji category, Unicode crosswalk

Description

A data set containing each Emoji category (such as Activities), its respective Unicodes string separated by |.

Usage

```
category_unicode_crosswalk
```

Format

A data frame with 10 rows and 2 columns:

category Emoji category (10 categories only)

unicodes The Unicodes string of Emojis belonging to category per se.

Source

The raw data set emojis comes from the emoji package, and it is processed by the author for the specific needs of tidyEmoji.

---

emoji_categorize | *Categorize Emoji Tweets/text based on Emoji category*

Description

Users can use `emoji_categorize` to see the all the categories each Emoji Tweet has. The function preserves the input data structure, and the only change is it adds an extra column with information about Emoji category separated by | if there is more than one category.

Usage

```
emoji_categorize(tweet_tbl, tweet_text)
```

Arguments

tweet_tbl   A dataframe/tibble containing tweets/text.

tweet_text   The tweet/text column.

Value

A filtered dataframe with the presence of Emoji only, and with an extra column `.emoji_category`.

Examples

```
library(dplyr)
data.frame(tweets = c("I love tidyverse \U0001f600\U0001f603\U0001f603",
                      "R is my language! \U0001f601\U0001f606\U0001f605",
                      "This Tweet does not have Emoji!",
                      "Wearing a mask\U0001f637\U0001f637\U0001f637.",
                      "Emoji does not appear in all Tweets",
                      "A flag \U0001f600\U0001f3c1")) %>%
        emoji_categorize(tweets)
```

---

emoji_extract_nest | *Emoji extraction nested summary*

Description

This function adds an extra list column called `.emoji_unicode` to the original data, with all Emojis included.

Usage

```
emoji_extract_nest(tweet_tbl, tweet_text)
```

Arguments

tweet_tbl	A dataframe/tibble containing tweets/text.
tweet_text	The tweet/text column.

Value

The original dataframe/tibble with an extra column collumn called `.emoji_unicode`.

Examples

```
library(dplyr)
data.frame(tweets = c("I love tidyverse \U0001f600\U0001f603\U0001f603",
                      "R is my language! \U0001f601\U0001f606\U0001f605",
                      "This Tweet does not have Emoji!",
                      "Wearing a mask\U0001f637\U0001f637\U0001f637.",
                      "Emoji does not appear in all Tweets",
                      "A flag \U0001f600\U0001f3c1")) %>%
        emoji_extract_nest(tweets)
```

---

emoji_extract_unnest   *Emoji extraction unnested summary*

Description

If users would like to know how many Emojis and what kinds of Emojis each Tweet has, `emoji_extract` is a useful function to output a global summary with the row number of each Tweet containing Emoji and the Unicodes associated with each Tweet.

Usage

```
emoji_extract_unnest(tweet_tbl, tweet_text)
```

Arguments

tweet_tbl	A dataframe/tibble containing tweets/text.
tweet_text	The tweet/text column.

Value

A summary tibble with the original row number and Emoji count.

Examples

```
library(dplyr)
data.frame(tweets = c("I love tidyverse \U0001f600\U0001f603\U0001f603",
                      "R is my language! \U0001f601\U0001f606\U0001f605",
                      "This Tweet does not have Emoji!",
                      "Wearing a mask\U0001f637\U0001f637\U0001f637.",
                      "Emoji does not appear in all Tweets",
                      "A flag \U0001f600\U0001f3c1")) %>%
        emoji_extract_unnest(tweets)
```

---

emoji_summary   *Emoji summary tibble*

Description

When having a Twitter dataframe/tibble at hand, it should be nice to know how many Tweets contain Emojis. This is the right time to use this function. What is worth noting is that it does not matter whether a Tweet has one Emoji or ten Emojis, the function only counts it once and returns a tibble that summarizes the number of Tweets containing at least one Emoji and the total number of Tweets presented in the dataframe/tibble.

Usage

```
emoji_summary(tweet_tbl, tweet_text)
```

Arguments

tweet_tbl   A dataframe/tibble containing tweets/text.

tweet_text   The tweet/text column.

Value

A summary tibble including # of Tweets in total and # of Tweets that have at least one Emoji.

Examples

```
library(dplyr)
data.frame(tweets = c("I love tidyverse \U0001f600\U0001f603\U0001f603",
                      "R is my language! \U0001f601\U0001f606\U0001f605",
                      "This Tweet does not have Emoji!",
                      "Wearing a mask\U0001f637\U0001f637\U0001f637.",
                      "Emoji does not appear in all Tweets",
                      "A flag \U0001f600\U0001f3c1")) %>%
        emoji_summary(tweets)
```

---

emoji_tweets   *Emoji Text/Tweets Output*

Description

When users just want to focus on Tweets containing Emoji(s), `emoji_tweets` filters out non-Emoji rows and only returns rows that have at least one Emoji.

Usage

```
emoji_tweets(tweet_tbl, tweet_text)
```

Arguments

tweet_tbl	A dataframe/tibble containing tweets/text.
tweet_text	The tweet/text column.

Value

A dataframe/tibble containing only text with at least one Emoji

Examples

```
library(dplyr)
data.frame(tweets = c("I love tidyverse \U0001f600\U0001f603\U0001f603",
                      "R is my language! \U0001f601\U0001f606\U0001f605",
                      "This Tweet does not have Emoji!",
                      "Wearing a mask\U0001f637\U0001f637\U0001f637.",
                      "Emoji does not appear in all Tweets",
                      "A flag \U0001f600\U0001f3c1")) %>%
        emoji_tweets(tweets)
```

---

emoji_unicode_crosswalk

Emoji name, Unicode, and Emoji category crosswalk

Description

A data set containing each Emoji name (such as grinning, smile), its respective Unicode and category. One thing to note here is there are duplicated Unicodes in the data set, because one Unicode could have multiple Emoji names.

Usage

```
emoji_unicode_crosswalk
```

Format

A data frame with 4536 rows and 3 columns:

emoji_name The name of Emoji per se.

unicode The Unicode of Emoji.

emoji_category The category Emoji falls into.

Source

The raw data sets (`emoji_name` and `emojis`) come from the `emoji` package, and they are processed by the author for the specific needs of `tidyEmoji`.

---

  tidyEmoji   tidyEmoji *package*

Description

A tidy way working with text containing Emoji.

---

  top_n_emojis   *Getting n most popular Emojis*

Description

When working with Tweets, counting how many times each Emoji appears in the entire Tweet corpus is useful. This is when `top_n_emojis` comes into play, and it is handy to see how Emojis are distributed across the corpus. If a Tweet has 10 Emojis, `top_n_emojis` will count it 10 times and assign each of the 10 Emojis on its respective Emoji category. What is interesting to note is Unicodes returned by `top_n_emojis` could have duplicates, meaning some Unicodes share various Emoji names. By default, this does not happen, but users can choose `duplicated_unicode = 'yes'` to obtain duplicated Unicodes.

Usage

```
top_n_emojis(tweet_tbl, tweet_text, n = 20, duplicated_unicode = "no")
```

Arguments

`tweet_tbl`	A dataframe/tibble containing tweets/text.
`tweet_text`	The tweet/text column.
`n`	Top n Emojis, default is 20.
`duplicated_unicode`	
	If no repetitious Unicode, no. Otherwise, yes. Default is no.

Value

A tibble with top n Emojis

Examples

```
library(dplyr)
data.frame(tweets = c("I love tidyverse \U0001f600\U0001f603\U0001f603",
                      "R is my language! \U0001f601\U0001f606\U0001f605",
                      "This Tweet does not have Emoji!",
                      "Wearing a mask\U0001f637\U0001f637\U0001f637.",
                      "Emoji does not appear in all Tweets",
                      "A flag \U0001f600\U0001f3c1")) %>%
         top_n_emojis(tweets, n = 2)
```

Index