

Package ‘trollR’

April 19, 2018

Title Detecting if a post is trolling, one post at a time

Version 0.0.1

Date 2018-04-18

Maintainer Gruppe 3 <email@zu.de>

Description What the package does (one paragraph).

Depends R (>= 3.4.3),
tidyverse,
shiny

Imports doParallel,
plumber,
xgboost,
text2vec,
tm,
Matrix,
tokenizers,
shinydashboard,
tokenizers

License MIT

Encoding UTF-8

LazyData true

Suggests testthat

RoxygenNote 6.0.1

R topics documented:

build_features	2
fake_evaluator	3
predict_troll	3
runExample	4
run_api	4
shitwordlist	5
test_function	5

Index	7
--------------	----------

build_features	<i>Builds the feature-matrix from a text-vector</i>
----------------	---

Description

Builds the feature-matrix from a text-vector

Usage

```
build_features(x, term_count_min = 1, mdl = NULL, parallel = TRUE,
              quiet = FALSE)
```

Arguments

x	a vector of text
term_count_min	a number passed to prune_vocabulary , defaults to 1. In case the function is used for training, it can and should be set to some higher value, i.e., 3.
mdl	is a list of existing models-data (containing the vectorizer, the tfidf, and the lsa object), defaults to NULL, in which case it is rebuild
parallel	T/F if the task should be executed in parallel, defaults to TRUE
quiet	T/F if the function remains silent, defaults to FALSE

Value

a list of two: a dgCMatrix that contains the features (columns) for each text (row) and as a second element a list of the model that can be passed as mdl

Examples

```
text <- c(
  "This is a first text that describes something",
  "A second Text That USES A LOT of CAPITALS",
  "Lastly MANY!!!! (like, really a lot!) punctuations!!!"
)

build_features(text)

dtm <- c("capit", "someth", "punctuat", "use", "mani", "second", "last",
        "describ", "like", "first", "realli")
build_features(text, dtm = dtm)

# a second example
#' train <- c("Banking is finance", "flowers are not houses", "finance is power", "houses are build")
test <- c("finance is greed", "flowers belong in the garbage", "houses are build")

a1 <- build_features(test)
a12 <- build_features(test, mdl = a1$mdl)

a2 <- build_features(train, mdl = a1$mdl)
a2$res
```

fake_evaluator	<i>This will be our evaluator function (later).</i>
----------------	---

Description

Takes in a string and classifies it.

Usage

```
fake_evaluator(string = "David kraut vajayjay.", testfor = c("toxic",  
  "severe_toxic", "obscene", "threat", "insult", "identity_hate"))
```

Arguments

string	A chracter string containing the bite to be classified.
--------	---

Value

What the function returns (eg. a data frame with n rows of v variables).

Warning

Do not operate heavy machinery within 8 hours of using this function.

Examples

```
fake_evaluator(string = "David kraut vajayjay")
```

predict_troll	<i>Detect if given texts are trolls</i>
---------------	---

Description

Detect if given texts are trolls

Usage

```
predict_troll(x, model_ = NULL, mdl_data_ = NULL)
```

Arguments

x	a vector of text
model_	a model that is passed to predict, defaults to the model supplied with this package
mdl_data_	a model as returned by build_features (the mdl) containing the vectorizer, tfidf, and the lsa objects. Defaults to the mdl_data from this package.

Value

a vector with the same lengths as x that holds the predicted probabilities that the given text is trolling

Examples

```
text <- c("You suck, die!", "What a nice world we have today", "I like you", "I hate you")
(pred <- predict_troll(text))
```

runExample	<i>Shiny App Launcher.</i>
------------	----------------------------

Description

Opens a shiny app. .

Usage

```
runExample(example = "trollR")
```

Arguments

example Name of the app.

Value

Opens the respective shiny app.

Warning

Do not operate heavy machinery within 8 hours of using this function. #' @export

Examples

```
runExample()
```

run_api	<i>Run the Plumber API</i>
---------	----------------------------

Description

Run the Plumber API

Usage

```
run_api(...)
```

Arguments

... parameters passed to [plumber](#)

Value

invisible NULL

Examples

```
## Not run:
  run_api()
  # try to got to: http://127.0.0.1:6327/trollR
  # or use http://127.0.0.1:6327/trollR?text=This may be a troll comment

## End(Not run)
```

shitwordlist	<i>Shitword list</i>
--------------	----------------------

Description

A list of English curse words scraped from <https://www.noswearing.com/dictionary>

Usage

```
shitwordlist
```

Format

A character vector containing 349 words.

test this is only for illustration purpose how to document data

london hello whatup

test_function	<i>The function name pipapo.</i>
---------------	----------------------------------

Description

What it does.

Usage

```
test_function(x = c("hello"), y = 2, z = c(2, 7, 9))
```

Arguments

x	A string vector.
y	An integer.
z	A numeric vector.

Value

What the function returns (eg. a data frame with n rows of v variables).

Warning

Do not operate heavy machinery within 8 hours of using this function.

Examples

```
test_function(x = c("hello"),  
             y = c(2),  
             z = c(2,7,9))
```

Index

- *Topic **datasets**
 - shitwordlist, [5](#)
- build_features, [2](#), [3](#)
- fake_evaluator, [3](#)
- plumber, [4](#)
- predict_troll, [3](#)
- prune_vocabulary, [2](#)
- run_api, [4](#)
- runExample, [4](#)
- shitwordlist, [5](#)
- test_function, [5](#)