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

Description What the package does (one paragraph).

Depends R (>= 3.4.3),
shiny,
tidyverse

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

License MIT

Encoding UTF-8

LazyData true

Suggests testthat

RoxygenNote 6.0.1

R topics documented:

build_features *Builds the feature-matrix from a text-vector*

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
```

predict_troll *Detect if given texts are trolls*

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))
```

run_api *Run the Plumber API*

Description

Run the Plumber API

Usage

```
run_api(port = 8000)
```

Arguments

...	parameters passed to `plumber`

Value

invisible NULL

Examples

```
## Not run:
  run_api()
  # try to got to: http://127.0.0.1:8000/trollR
  # or use http://127.0.0.1:8000/trollR?text=This may be a troll comment

## End(Not run)
```

`run_shiny` *Shiny App Launcher*

Description

Shiny App Launcher

Usage

```
run_shiny(example = "trollR")
```

Arguments

`example`	name of the app, defaults to trollR

Value

Nothing

Examples

```
## Not run:
 run_shiny()

## End(Not run)
```

`shitwordlist` *Shitword list*

Description

A list of Enlish 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

Index