

# Package ‘needmining’

March 8, 2019

**Type** Package

**Title** A Simple Needmining Implementation

**Version** 0.1.1

**Author** Dorian Proksch <dorian.proksch@hh1.de>, Timothy P. Jurka [ctb], Yoshimasa Tsuruoka [ctb], Loren Collingwood [ctb], Amber E. Boydston [ctb], Emiliano Grossman [ctb], Wouter van Atteveldt [ctb]

**Maintainer** Dorian Proksch <dorian.proksch@hh1.de>

**Description** Showcasing needmining (the semi-automatic extraction of customer needs from social media data) with Twitter data. It uses the handling of the Twitter API provided by the package 'rtweet' and the textmining algorithms provided by the package 'tm'.

Niklas Kuehl (2016) <doi:10.1007/978-3-319-32689-4\_14> wrote an introduction to the topic of needmining.

**License** GPL-3

**Imports** methods, utils, rtweet, randomForest, stringr, SnowballC, SparseM, tau, tm

**Encoding** UTF-8

**RoxygenNote** 6.1.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2019-03-08 19:20:03 UTC

## R topics documented:

needmining-package	2
downloadTweets	2
filterTweetsMachineLearning	3
filterTweetsNeedwords	4
NMdataToClassify	5
NMTrainingData	5
readNeedminingFile	6
removeTweetsStopwords	6
saveNeedminingFile	7
twitterLogin	8

**Index****9**


---

needmining-package	<i>Functions for a simple needmining implementation</i>
--------------------	---

---

**Description**

needmining provides the basic functionality to download social media data from Twitter and semi automatically classify the data regarding user needs

---

downloadTweets	<i>Downloading Tweets based on a keyword list</i>
----------------	---

---

**Description**

downloadTweets downloads Tweets containing specified keywords from the Twitter API

**Usage**

```
downloadTweets(search_terms, n = 100, lang = "en")
```

**Arguments**

search_terms	a string containing the search terms in Twitter format (use OR and AND to connect multiple search terms in one search)
n	The number of Tweets downloaded. Please note that this limit is based on your Twitter account
lang	The language of the Tweets. Default is English. Please refer to the Twitter API documentation for language codes

**Details**

This function downloads Tweets for a specified keyword list, removes line breaks, adds a column isNeed filled with 0

**Value**

a data frame containing the tweets as well as an additional column isNeed filled with 0

**Author(s)**

Dorian Proksch <dorian.proksch@hhl.de>

**Examples**

```
searchterm <- "smart speaker" OR "homepod" OR "google home mini"  
## Not run:  
token <- twitterLogin()  
currentTweets <- downloadTweets(searchterm, n = 180)  
  
## End(Not run)
```

---

filterTweetsMachineLearning

*Classify needs based on machine learning*

---

**Description**

filterTweetsMachineLearning classifies a list of Tweets as needs based on the random forest machine learning algorithm

**Usage**

```
filterTweetsMachineLearning(dataToClassify, trainingData)
```

**Arguments**

`dataToClassify` a dataframe containing the Tweet messages to classify  
`trainingData` a dataframe containing Tweets messages with a given classification (0=not a need, 1=a need)

**Details**

This function uses a machine learning algorithm (random forest) to classify needs based on their content. It needs a training data set with classified needs (indicated by 0=not a need, 1=a need). This function used code fragments from the archived R packages maxent and RTextTools. The authors are Timothy P. Jurka, Yoshimasa Tsuruoka, Loren Collingwood, Amber E. Boydston, Emiliano Grossman, Wouter van Atteveldt

**Value**

a dataframe with classified data

**Author(s)**

Dorian Proksch <dorian.proksch@hhl.de>

## Examples

```
data(NMTrainingData)
data(NMdataToClassify)
smallNMTrainingData <- rbind(NMTrainingData[1:75,], NMTrainingData[101:175,])
smallNMdataToClassify <- rbind(NMdataToClassify[1:10,], NMdataToClassify[101:110,])
results <- filterTweetsMachineLearning(smallNMdataToClassify, smallNMTrainingData)
```

---

filterTweetsNeedwords *Filter tweets containing need indicating words*

---

## Description

filterTweetsNeedwords filters a list of Tweets regarding need indicating words

## Usage

```
filterTweetsNeedwords(tweetMessages, needWords)
```

## Arguments

tweetMessages a dataframe containing the Tweet messages  
needWords a string containing needwords separately by `;`

## Details

This function filters Tweets regarding a list of need indicating words

## Value

a filtered data frame

## Author(s)

Dorian Proksch <dorian.proksch@hhl.de>

## Examples

```
data(NMTrainingData)
needWordsNeedsOnly <- "need;want;wish;feature;ask;would like;improve;idea;upgrade"
needsSimple <- filterTweetsNeedwords(NMTrainingData, needWordsNeedsOnly)
needWordsExtended <- "need;want;wish;feature;ask;would like;improve;idea;upgrade;
support;problem;issue;help;fix;complain;fail"
needsSimpleExtended <- filterTweetsNeedwords(NMTrainingData, needWordsExtended)
```

---

NMdataToClassify	<i>Test dataset regarding the user needs for smart speakers</i>
------------------	---

---

**Description**

A dataset containing 200 artificially generated messages in the Twitter format for the topic of smart speakers. These messages are inspired by real Tweets (rephrased, anonymized, all brand names removed). Furthermore, Tweets containing stopwords were removed. 100 rows contain user needs, 100 rows contain no user needs. The data is coded (0=no need,1=a need). It can be used to test a classification algorithm.

**Usage**

```
data(NMdataToClassify)
```

**Format**

A data frame with 200 rows and 2 variables:

**Tweets** Contains the message

**isNeed** Is a need described within the message? 0=no, 1=yes

---

NMTrainingData	<i>Training dataset regarding the user needs for smart speakers</i>
----------------	---

---

**Description**

A dataset containing 200 artificially generated messages in the Twitter format for the topic of smart speakers. These messages are inspired by real Tweets (rephrased, anonymized, all brand names removed). 100 rows contain user needs, 100 rows contain no user needs. The data is coded (0=no need,1=a need). The data can be used to train a classification algorithm.

**Usage**

```
data(NMTrainingData)
```

**Format**

A data frame with 200 rows and 2 variables:

**Tweets** Contains the message

**isNeed** Is a need described within the message? 0=no, 1=yes

readNeedminingFile *Read Tweet file*

---

**Description**

readNeedminingFile reads a Needmining file created by the needmining package

**Usage**

```
readNeedminingFile(filename)
```

**Arguments**

filename            The filename of the file to read

**Details**

This function reads a Needmining file created by the needmining package

**Value**

a data frame containing the content of the file

**Author(s)**

Dorian Proksch <dorian.proksch@hhl.de>

**Examples**

```
data(NMTrainingData)
saveNeedminingFile(filename=file.path(tempdir(), "NMTrainingData.csv"),
  NMTrainingData)
currentNeedData <- readNeedminingFile(file.path(tempdir(), "NMTrainingData.csv"))
```

---

removeTweetsStopwords *Remove Tweets containing stopwords*

---

**Description**

removeTweetsStopwords removes Tweets containing stopwords

**Usage**

```
removeTweetsStopwords(tweetMessages, stopWords)
```

**Arguments**

tweetMessages a dataframe containing the Tweet messages  
stopWords a string containing stopwords separated by ','

**Details**

This function removes Tweets containing stopwords from a list of Twitter messages.

**Value**

a filtered data frame

**Author(s)**

Dorian Proksch <dorian.proksch@hhl.de>

**Examples**

```
stopWords <- "review;giveaway;save;deal;win;won;price;launch;news;gift;announce;  
  reveal;sale;http;buy;bought;purchase;sell;sold;invest;discount;  
  coupon;ship;giving away"  
data(NMTrainingData)  
filteredTweets <- removeTweetsStopwords(NMTrainingData, stopWords)
```

---

saveNeedminingFile *Save Tweet file*

---

**Description**

saveNeedminingFile saves a dataframe created by the needmining package to a file

**Usage**

```
saveNeedminingFile(filename, tweetMessages)
```

**Arguments**

filename The filename to save to  
tweetMessages An object containing the Twitter messages

**Details**

This function saves a dataframe created by the needmining package to a file

**Author(s)**

Dorian Proksch <dorian.proksch@hhl.de>

**Examples**

```
data(NMTrainingData)
saveNeedminingFile(filename=file.path(tempdir(), "NMTrainingData.csv"),
NMTrainingData)
```

---

twitterLogin

*Login into Twitter API*

---

**Description**

twitterLogin creates a token for the Twitter API

**Usage**

```
twitterLogin()
```

**Details**

This function creates a Twitter token of the Twitter API. This is necessary to use functions of the Twitter API. The login data has to be stored in the 'TwitterLoginData.csv' in the current set working directory (please refer to `getwd()` and `setwd()`). The file should have the following format: START app;consumer\_key;consumer\_secret;access\_token;access\_secret LINEBREAK The name of your app; your consumer\_key; your consumer\_secret; your access\_token; your access\_secret END OF FILE

**Value**

a Twitter token

**Author(s)**

Dorian Proksch <dorian.proksch@hhl.de>

**Examples**

```
## Not run:
token <- twitterLogin()

## End(Not run)
```



# Index

## \*Topic **datasets**

NMdataToClassify, [5](#)

NMTrainingData, [5](#)

downloadTweets, [2](#)

filterTweetsMachineLearning, [3](#)

filterTweetsNeedwords, [4](#)

needmining (needmining-package), [2](#)

needmining-package, [2](#)

NMdataToClassify, [5](#)

NMTrainingData, [5](#)

readNeedminingFile, [6](#)

removeTweetsStopwords, [6](#)

saveNeedminingFile, [7](#)

twitterLogin, [8](#)