Package 'newsmap'

December 12, 2019

Type Package

Title Semi-Supervised Model for Geographical Document Classification

Version 0.7.0

Maintainer Kohei Watanabe <watanabe.kohei@gmail.com>

Description Semi-supervised model for geographical document classification (Watanabe 2018) <doi:10.1080/21670811.2017.1293487>.
This package currently contains seed dictionaries in English, German, French, Spanish, Russian, Hebrew, Arabic Japanese and Chinese (Simplified and Traditional).

License MIT + file LICENSE

URL https://github.com/koheiw/newsmap

BugReports https://github.com/koheiw/newsmap/issues

LazyData TRUE

Encoding UTF-8

Depends R (>= 3.5), methods

Imports utils, Matrix, quanteda (>= 1.4), stringi,

Suggests testthat,

RoxygenNote 7.0.2

NeedsCompilation no

Author Kohei Watanabe [aut, cre, cph],
Stefan Müller [aut],
Dani Madrid-Morales [aut],
Katerina Tertytchnaya [aut],
Ke Cheng [aut],
Chung-hong Chan [aut],
Claude Grasland [aut],
Giuseppe Carteny [aut],
Elad Segev [aut],
Dai Yamao [aut]

Repository CRAN

Date/Publication 2019-12-11 23:50:02 UTC

R topics documented:

`accuracy` *Evaluate classification accuracy in precision and recall*

Description

Evaluate classification accuracy in precision and recall

Usage

```
accuracy(x, y)
```

Arguments

`x`	vercor of predicted classes
`y`	vector of true classes

Examples

```
class_pred <- c('US', 'GB', 'US', 'CN', 'JP', 'FR', 'CN') # prediction
class_true <- c('US', 'FR', 'US', 'CN', 'KP', 'EG', 'US') # true class
acc <- accuracy(class_pred, class_true)
print(acc)
summary(acc)
```

afe *Compute average feature entroy*

Description

Compute average feature entroy

Usage

```
afe(x, y, smooth = 1)
```

Arguments

x	a dfm for features
y	a dfm for labels
smooth	a numeric value for smoothing to include all the features

data_dictionary_newsmap_ar *Seed geographical dictionary in Arabic*

Description

Seed geographical dictionary in Arabic

Author(s)

Dai Yamao <daiyamao@scs.kyushu-u.ac.jp>

data_dictionary_newsmap_de *Seed geographical dictionary in German*

Description

Seed geographical dictionary in German

Author(s)

Stefan Müller <mullers@tcd.ie>

`data_dictionary_newsmap_en`
Seed geographical dictionary in English

Description

Seed geographical dictionary in English

Author(s)

Kohei Watanabe <watanabe.kohei@gmail.com>

`data_dictionary_newsmap_es`
Seed geographical dictionary in Spanish

Description

Seed geographical dictionary in Spanish

Author(s)

Dani Madrid-Morales <dani.madrid@my.cityu.edu.hk>

`data_dictionary_newsmap_fr`
Seed geographical dictionary in French

Description

Seed geographical dictionary in French

Author(s)

Claude Grasland <claude.grasland@parisgeo.cnrs.fr>

data_dictionary_newsmap_he

Seed geographical dictionary in Hebrew

Description

Seed geographical dictionary in Hebrew

Author(s)

Elad Segev <eladseg@gmail.com>

data_dictionary_newsmap_it

Seed geographical dictionary in Italian

Description

Seed geographical dictionary in Italian

Author(s)

Giuseppe Carteny <giuseppe.carteny@unimi.it>

data_dictionary_newsmap_ja

Seed geographical dictionary in Japanese

Description

Seed geographical dictionary in Japanese

Author(s)

Kohei Watanabe <watanabe.kohei@gmail.com>

`data_dictionary_newsmap_ru`
Seed geographical dictionary in Russian

Description

Seed geographical dictionary in Russian

Author(s)

Katerina Tertytchnaya <katerina.tertytchnaya@gmail.com>

`data_dictionary_newsmap_zh_cn`
Seed geographical dictionary in Chinese (simplified)

Description

Seed geographical dictionary in Chinese (simplified)

Author(s)

Ke Cheng <kecheng.ac@gmail.com>

`data_dictionary_newsmap_zh_tw`
Seed geographical dictionary in Chinese (traditional)

Description

Seed geographical dictionary in Chinese (traditional)

Author(s)

Chung-hong Chan <chainsawtiney@gmail.com>

predict.textmodel_newsmap

Prediction method for textmodel_newsmap

Description

Predict document class using trained a Newsmap model

Usage

```
## S3 method for class 'textmodel_newsmap'
predict(
  object,
  newdata = NULL,
  confidence.fit = FALSE,
  rank = 1L,
  type = c("top", "all"),
  ...
)
```

Arguments

object	a fitted Newsmap textmodel
newdata	dfm on which prediction should be made
confidence.fit	if TRUE, likelihood ratio score will be returned
rank	rank of class to be predicted. Only used when type = "top".
type	if top, return the most likely class specified by rank; otherswise return a matrix of lilelyhood ratio scores for all possible classes
...	not used.

print.textmodel_newsmap_summary

Print method for a fitted Newsmap model

Description

Print method for a fitted Newsmap model

Usage

```
## S3 method for class 'textmodel_newsmap_summary'
print(x, ...)
```

Arguments

x	a fitted Newsmap textmodel
...	not used.

`summary.textmodel_newsmap_accuracy` *Calcualte micro and macro average measures of accuracy*

Description

This function calculates micro-averave precision (p) and recall (r) and macro-average precision (P) and recall (R) based on a confusion matrix from `accuracy()`.

Usage

```
## S3 method for class 'textmodel_newsmap_accuracy'
summary(object, ...)
```

Arguments

object	output of accuracy()
...	not used.

`textmodel_newsmap` *Semi-supervised Bayesian multinomial model for geographical document classification*

Description

Train a Newsmap model to predict geographical focus of documents using a pre-defined seed dictionary. Currently seed dictionaries are available in English (en), German (de), Spanish (es), Japanese (ja), Russian (ru) and Chinese (zh).

Usage

```
textmodel_newsmap(x, y, smooth = 1, verbose = quanteda_options("verbose"))
```

Arguments

x	dfm from which features will be extracted
y	dfm in which features will be class labels
smooth	smoothing parameter for word frequency
verbose	if TRUE, show progress of training

References

Kohei Watanabe. 2018. "Newsmap: semi-supervised approach to geographical news classification." *Digital Journalism* 6(3): 294-309.

Examples

```
require(quanteda)
text_en <- c(text1 = "This is an article about Ireland.",
             text2 = "The South Korean prime minister was re-elected.")

toks_en <- tokens(text_en)
label_toks_en <- tokens_lookup(toks_en, data_dictionary_newsmap_en, levels = 3)
label_dfm_en <- dfm(label_toks_en)

feat_dfm_en <- dfm(toks_en, tolower = FALSE)

model_en <- textmodel_newsmap(feat_dfm_en, label_dfm_en)
predict(model_en)
```

Index