

Package ‘textmatch’

April 1, 2019

Title Toolkit for Matching Textual Data and Evaluating Textual Similarity

Version 0.0.0.9000

Description What the package does (one paragraph).

Depends R (>= 3.5.2)

License What license is it under?

Encoding UTF-8

LazyData false

RoxygenNote 6.1.1.9000

Imports dplyr,
data.table,
quantda

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

FoxCNNcorpus	2
FoxCNNmeta	2
FoxCNNscores	3
FoxCNNsurvey	3
get_similarity_scores	4
makeMatches	4
pair_distances	5
textmatch	5
transform_dfm	6
transform_stm	7
Index	8

FoxCNNcorpus	<i>Corpus with 1,565 articles from CNN and 1,796 articles from Fox News.</i>
--------------	--

Description

Corpus with 1,565 articles from CNN and 1,796 articles from Fox News.

Usage

FoxCNNcorpus

Format

A [data.frame](#) with 3,361 observations of 5 articles

Details

Corpus of front-page news articles published online by CNN or Fox News from 12/20/2014 to 05/09/2015 containing 1,565 articles from CNN and 1,796 articles from Fox News. Data contains article identifiers corresponding to data in FoxCNNmeta as well as raw and cleaned text data.

References

Mozer et al. (2019) [Matching with Text Data: An Experimental Evaluation of Methods for Matching Documents and of Measuring Match Quality](#). *Political Analysis*, Forthcoming.

FoxCNNmeta	<i>Dataset containing metadata for FoxCNN corpus with 1,565 articles from CNN and 1,796 articles from Fox News.</i>
------------	---

Description

Dataset containing metadata for FoxCNN corpus with 1,565 articles from CNN and 1,796 articles from Fox News.

Usage

FoxCNNmeta

Format

A [data.frame](#) with 3,361 observations of 5 articles

Details

Metadata for the FoxCNN corpus including article names, original URLs, and dates of publication.

References

Mozer et al. (2019) [Matching with Text Data: An Experimental Evaluation of Methods for Matching Documents and of Measuring Match Quality](#). *Political Analysis*, Forthcoming.

FoxCNNscores	<i>Dataset containing quality scores for 2.8 million possible matched pairs of Fox and CNN articles corresponding to 117 different combinations of distance metric and text representation, enumerated in Mozer et al. (2019), Section 3. Pair quality scores are estimated from a predictive model trained on a sample of 505 pairs evaluated by human coders.</i>
--------------	---

Description

Dataset containing quality scores for 2.8 million possible matched pairs of Fox and CNN articles corresponding to 117 different combinations of distance metric and text representation, enumerated in Mozer et al. (2019), Section 3. Pair quality scores are estimated from a predictive model trained on a sample of 505 pairs evaluated by human coders.

Usage

FoxCNNscores

Format

A [data.frame](#) with 2,810,740 pairs and 117 features.

Details

Aggregated quality scores for all 2.8 million possible pairs of Fox and CNN articles based on the predictive model trained on ratings from human coders.

References

Mozer et al. (2019) [Matching with Text Data: An Experimental Evaluation of Methods for Matching Documents and of Measuring Match Quality](#). *Political Analysis*, Forthcoming.

FoxCNNsurvey	<i>Dataset containing distance measurements and average quality scores for sample of 505 pairs of matched Fox and CNN articles evaluated by human coders on Mechanical Turk. Raw quality scores are regressed on distance measurements to fit a predictive model for match quality as a function of the 117 metrics considered.</i>
--------------	---

Description

Dataset containing distance measurements and average quality scores for sample of 505 pairs of matched Fox and CNN articles evaluated by human coders on Mechanical Turk. Raw quality scores are regressed on distance measurements to fit a predictive model for match quality as a function of the 117 metrics considered.

Usage

FoxCNNsurvey

Format

A [data.frame](#) with 505 pairs and 108 features.

Details

Distance measurements and average quality scores for a sample of 505 matched pairs of documents evaluated by human coders.

References

Mozer et al. (2019) **Matching with Text Data: An Experimental Evaluation of Methods for Matching Documents and of Measuring Match Quality**. *Political Analysis*, Forthcoming.

get_similarity_scores	<i>This function calculates an input character vector's similarity matrix according to the measures contained in the predictive model.</i>
-----------------------	--

Description

This function calculates an input character vector's similarity matrix according to the measures contained in the predictive model.

Usage

```
get_similarity_scores(x)
```

Arguments

x	A character vector where each element is a document
---	---

Value

A data frame of rows ($n * n-1$) and columns 16; each column is one of the constituent similarity measures

makeMatches	<i>Create a data frame of matched pairs of documents and return indices for matched sets</i>
-------------	--

Description

Create a data frame of matched pairs of documents and return indices for matched sets

Usage

```
makeMatches(match.obj, Z)
```

Arguments

match.obj	a matched data set
Z	a vector of treatment indicators

Value

A [data.frame](#) of indices for matched pairs of documents

pair_distances	<i>Similarity and distance computation between documents or features</i>
----------------	--

Description

These functions compute distance matrices from a text representation where each row is a document and each column is a feature to measure distance over based on treatment indicator Z

Usage

```
pair_distances(dat, Z, include = c("cosine", "euclidean", "mahalanobis",
  "propensity"), propensity.method = c("glm", "mnir"),
  exclude = "jaccard", docnames = NULL, verbose = TRUE)
```

Arguments

Z	A logical or binary vector indicating treatment and control for each unit in the study. TRUE or 1 represents a treatment unit, FALSE or 0 represents a control unit.
propensity.method	Either GLM or MNIR for propensity score estimation
docnames	A vector of document names equal in length to the number of documents
x	a matrix text representation with rows corresponding to each document in a corpus and columns that represent summary measures of the text (e.g., word counts, topic proportions, etc.). Acceptable forms include a valid quanteda dfm object, a tm Document-Term Matrix, or a matrix of estimated topic proportions.

Value

A matrix showing pairwise distances for all potential matches of treatment and control units under various distance metrics

textmatch	<i>This function runs the main ML model as specified in Mozer et al. (2018)</i>
-----------	---

Description

This function runs the main ML model as specified in Mozer et al. (2018)

Usage

```
textmatch(x, outcome = "matrix")
```

Arguments

`x` A character vector where each element is a document

Value

An n by n matrix where n is the length of parameter `x`. Each entry is a standardized similarity score.

Examples

```
textmatch(c("I am a dog", "I am a cat", "The rain in Spain falls mainly on the plain."),
  outcome = "matrix")
```

<code>transform_dfm</code>	<i>Applies bounds, weights, and/or coarsening schemes to a dfm or document frequency matrix to reduce the dimension of the data, reduce noise, or apply other design rules (e.g. - to exclude words that occur in too few or too many documents).</i>
----------------------------	---

Description

Applies bounds, weights, and/or coarsening schemes to a dfm or document frequency matrix to reduce the dimension of the data, reduce noise, or apply other design rules (e.g. - to exclude words that occur in too few or too many documents).

Usage

```
transform_dfm(x, bounds = c(2, nrow(x) - 1), tfidf = FALSE,
  verbose = TRUE)
```

Arguments

<code>x</code>	a matrix text representation with rows corresponding to each document in a corpus and columns that represent summary measures of the text (e.g., word counts, topic proportions, etc.). Acceptable forms include a valid quanteda dfm object, a tm Document-Term Matrix, or a matrix of estimated topic proportions.
<code>bounds</code>	a vector of lower and upper bounds to enforce. Defaults to excluding any terms that appear in only one document and any terms that appear in every document
<code>tfidf</code>	optional scheme to use for weighting the DTM. Defaults to FALSE.
<code>verbose</code>	indicator for verbosity

Value

A bounded DFM

transform_stm	<i>Refits a STM with a content-based covariate so that all document-level topic-proportions are estimated "as-treated". Also allows for calculation of the SR sufficient reduction and optional coarsening to reduce the dimension of the data, reduce noise, or apply other design rules (e.g. - to exclude words that occur in too few or too many documents).</i>
---------------	--

Description

Refits a STM with a content-based covariate so that all document-level topic-proportions are estimated "as-treated". Also allows for calculation of the SR sufficient reduction and optional coarsening to reduce the dimension of the data, reduce noise, or apply other design rules (e.g. - to exclude words that occur in too few or too many documents).

Usage

```
transform_stm(mod, out, Z, calc.SR = TRUE, coarsen = FALSE)
```

Arguments

mod	a fitted stm object
out	the original call to the STM
Z	an indicator for treatment assignment
calc.SR	an indicator for returning the sufficient reduction. Default is TRUE.
coarsen	an indicator for returning the coarsened STM

Value

A bounded DFM

Index

*Topic **data**

FoxCNNcorpus, [2](#)

FoxCNNmeta, [2](#)

FoxCNNscores, [3](#)

FoxCNNsurvey, [3](#)

data.frame, [2–5](#)

FoxCNNcorpus, [2](#)

FoxCNNmeta, [2](#)

FoxCNNscores, [3](#)

FoxCNNsurvey, [3](#)

get_similarity_scores, [4](#)

makeMatches, [4](#)

pair_distances, [5](#)

stm, [7](#)

textmatch, [5](#)

transform_dfm, [6](#)

transform_stm, [7](#)