Package 'seededlda'

April 8, 2021

Type Package

Title Seeded-LDA for Topic Modeling

Version 0.6.0

Description Implements the seeded-
LDA model (Lu, Ott, Cardie & Tsou 2010) <doi:10.1109/ICDMW.2011.125> us-
ing the quanteda package and the GibbsLDA++ library for semisupervised topic modeling.
Seeded-LDA allows users to pre-define topics with keywords to perform theory-driven analy-
sis of textual data in social sciences and humanities (Watan-
abe & Zhou 2020) <doi:10.1177/0894439320907027>.

License GPL-3

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

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

Encoding UTF-8

Depends R (>= 3.5.0), quanteda (> 2.0), methods

Imports Matrix

LinkingTo Rcpp, RcppParallel, RcppArmadillo (>= 0.7.600.1.0), quanteda

Suggests testthat, quanteda.textmodels, topicmodels

RoxygenNote 7.1.1

NeedsCompilation yes

Author Kohei Watanabe [aut, cre, cph],
Phan Xuan-Hieu [aut, cph] (GibbsLDA++)

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

Repository CRAN

Date/Publication 2021-04-08 05:00:02 UTC

R topics documented:

`predict.textmodel_lda` *Prediction method for textmodel_lda*

Description

Predicts topics of documents with a fitted LDA model. Prediction is performed by a Gibbs sampling with words allocated to topics in the fitted LDA. The result becomes different from `topics()` even for the same documents because `predict()` triggers additional iterations.

Usage

```
## S3 method for class 'textmodel_lda'
predict(
  object,
  newdata = NULL,
  max_iter = 2000,
  verbose = quanteda_options("verbose"),
  ...
)
```

Arguments

`object`	a fitted LDA textmodel
`newdata`	dfm on which prediction should be made
`max_iter`	the maximum number of iteration in Gibbs sampling.
`verbose`	logical; if `TRUE` print diagnostic information during fitting.
`...`	not used

References

Lu, Bin et al. (2011). "Multi-aspect Sentiment Analysis with Topic Models". doi:10.5555/2117693.2119585. *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*.

Watanabe, Kohei & Zhou, Yuan (2020). "Theory-Driven Analysis of Large Corpora: Semisupervised Topic Classification of the UN Speeches". doi:10.1177/0894439320907027. *Social Science Computer Review*.

See Also

topicmodels

Examples

```
## Not run:
require(quanteda)

data("data_corpus_moviereviews", package = "quanteda.textmodels")
corp <- head(data_corpus_moviereviews, 500)
toks <- tokens(corp, remove_punct = TRUE, remove_symbols = TRUE, remove_number = TRUE)
dfmt <- dfm(toks) %>%
    dfm_remove(stopwords('en'), min_nchar = 2) %>%
    dfm_trim(min_termfreq = 0.90, termfreq_type = "quantile",
             max_docfreq = 0.1, docfreq_type = "prop")

# unsupervised LDA
lda <- textmodel_lda(head(dfmt, 450), 6)
terms(lda)
topics(lda)
predict(lda, newdata = tail(dfmt, 50))

# semisupervised LDA
dict <- dictionary(list(people = c("family", "couple", "kids"),
                        space = c("alien", "planet", "space"),
                        moster = c("monster*", "ghost*", "zombie*"),
                        war = c("war", "soldier*", "tanks"),
                        crime = c("crime*", "murder", "killer")))
slda <- textmodel_seededlda(dfmt, dict, residual = TRUE, min_termfreq = 10)
terms(slda)
topics(slda)

## End(Not run)
```

---

terms *Extract most likely terms*

Description

Extract most likely terms

Usage

```
terms(x, n = 10)
```

Arguments

x	a fitted LDA model
n	number of terms to be extracted

textmodel_lda *Semisupervised Latent Dirichlet allocation*

Description

`textmodel_seededlda()` implements semisupervised Latent Dirichlet allocation (seeded-LDA). The estimator's code adopted from the GibbsLDA++ library (Xuan-Hieu Phan, 2007). `textmodel_seededlda()` allows identification of pre-defined topics by semisupervised learning with a seed word dictionary.

Usage

```
textmodel_lda(
  x,
  k = 10,
  max_iter = 2000,
  alpha = NULL,
  beta = NULL,
  verbose = quanteda_options("verbose")
)

textmodel_seededlda(
  x,
  dictionary,
  valuetype = c("glob", "regex", "fixed"),
  case_insensitive = TRUE,
  residual = FALSE,
  weight = 0.01,
  max_iter = 2000,
  alpha = NULL,
  beta = NULL,
  ...,
  verbose = quanteda_options("verbose")
)
```

Arguments

x	the dfm on which the model will be fit
k	the number of topics
max_iter	the maximum number of iteration in Gibbs sampling.
alpha	the hyper parameter for topic-document distribution
beta	the hyper parameter for topic-word distribution
verbose	logical; if TRUE print diagnostic information during fitting.
dictionary	a quanteda::dictionary() with seed words that define topics.
valuetype	see quanteda::valuetype

case_insensitive	see quanteda::valuetype
residual	if TRUE a residual topic (or "garbage topic") will be added to user-defined topics.
weight	pseudo count given to seed words as a proportion of total number of words in x.
...	passed to quanteda::dfm_trim to restrict seed words based on their term or document frequency. This is useful when glob patterns in the dictionary match too many words.

References

Lu, Bin et al. (2011). "Multi-aspect Sentiment Analysis with Topic Models". doi:10.5555/2117693.2119585. *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*.

Watanabe, Kohei & Zhou, Yuan (2020). "Theory-Driven Analysis of Large Corpora: Semisupervised Topic Classification of the UN Speeches". doi:10.1177/0894439320907027. *Social Science Computer Review*.

See Also

topicmodels

Examples

```
## Not run:
require(quanteda)

data("data_corpus_moviereviews", package = "quanteda.textmodels")
corp <- head(data_corpus_moviereviews, 500)
toks <- tokens(corp, remove_punct = TRUE, remove_symbols = TRUE, remove_number = TRUE)
dfmt <- dfm(toks) %>%
    dfm_remove(stopwords('en'), min_nchar = 2) %>%
    dfm_trim(min_termfreq = 0.90, termfreq_type = "quantile",
             max_docfreq = 0.1, docfreq_type = "prop")

# unsupervised LDA
lda <- textmodel_lda(head(dfmt, 450), 6)
terms(lda)
topics(lda)
predict(lda, newdata = tail(dfmt, 50))

# semisupervised LDA
dict <- dictionary(list(people = c("family", "couple", "kids"),
                        space = c("alien", "planet", "space"),
                        moster = c("monster*", "ghost*", "zombie*"),
                        war = c("war", "soldier*", "tanks"),
                        crime = c("crime*", "murder", "killer")))
slda <- textmodel_seededlda(dfmt, dict, residual = TRUE, min_termfreq = 10)
terms(slda)
topics(slda)

## End(Not run)
```

topics *Extract most likely topics*

Description

Extract most likely topics

Usage

```
topics(x)
```

Arguments

x	a fitted LDA model

Index