Package 'keyATM'

April 15, 2020

Version 0.1.0

Title Keyword Assisted Topic Model

Description
Fits keyword assisted topic models (keyATM) using collapsed Gibbs samplers. The keyATM combines the latent dirichlet allocation (LDA) models with a small number of keywords selected by researchers in order to improve the interpretability and topic classification of the LDA. The keyATM can also incorporate covariates and directly model time trends. The keyATM is proposed in Eshima, Imai, and Sasaki (2020) <arXiv:2004.05964>.

License GPL-3

Depends R (>= 3.5)

Imports Rcpp, dplyr, fastmap, ggplot2, ggrepel, magrittr, Matrix, parallel, purrr, quanteda, rlang, stats, stringr, tibble, tidyr

LinkingTo Rcpp, RcppEigen, RcppProgress

Suggests readtext

URL https://keyatm.github.io/keyATM/

Encoding UTF-8

BugReports https://github.com/keyATM/keyATM/issues

LazyData TRUE

RoxygenNote 7.1.0

SystemRequirements C++11

NeedsCompilation yes

Author Shusei Eshima [aut, cre],
Tomoya Sasaki [aut],
William Lowe [ctb],
Kosuke Imai [aut]

Maintainer Shusei Eshima <shuseieshima@g.harvard.edu>

Repository CRAN

Date/Publication 2020-04-15 12:50:06 UTC

R topics documented:

`keyATM-package` *Keyword Assisted Topic Models*

Description

The implementation of keyATM models.

Author(s)

Maintainer: Shusei Eshima <shuseieshima@g.harvard.edu>

Authors:

- Tomoya Sasaki <tomoyas@mit.edu>
- Kosuke Imai <imai@harvard.edu>

Other contributors:

- William Lowe <wlowe@princeton.edu> [contributor]

See Also

Useful links:

- https://keyatm.github.io/keyATM/
- Report bugs at https://github.com/keyATM/keyATM/issues

by_strata_DocTopic *Estimate document-topic distribution by strata (for covariate models)*

Description

Estimate document-topic distribution by strata (for covariate models)

Usage

```
by_strata_DocTopic(
  x,
  by_name,
  by_values,
  burn_in = NULL,
  parallel = TRUE,
  mc.cores = NULL,
  posterior_mean = FALSE
)
```

Arguments

x	the output from a keyATM model (see keyATM())
by_name	character. The name of the variable to use.
by_values	numeric. The values of the variable specified in 'by_name'
burn_in	integer. Burn-in period. If not specified, it is the half of samples. Default is NULL.
parallel	logical. If TRUE, parallelization for speeding up. Default is TRUE.
mc.cores	integer. The number of cores to use. Default is NULL.
posterior_mean	logical. If TRUE, the quantity of interest to estimate is the posterior mean. Default is FALSE.

Value

strata_topicword object (a list)

by_strata_TopicWord *Estimate subsetted topic-word distribution*

Description

Estimate subsetted topic-word distribution

Usage

```
by_strata_TopicWord(x, keyATM_docs, by)
```

Arguments

x	the output from a keyATM model (see keyATM())
keyATM_docs	an object generated by keyATM_read() (see keyATM_read())
by	a vector whose length is the number of documents

Value

strata_topicword object (a list)

---

keyATM *keyATM main function*

Description

Run keyATM models.

Usage

```
keyATM(
  docs,
  model,
  no_keyword_topics,
  keywords = list(),
  model_settings = list(),
  priors = list(),
  options = list(),
  keep = c()
)
```

Arguments

docs	texts read via keyATM_read()
model	keyATM model: "base", "covariates", "dynamic", and "label"
no_keyword_topics	the number of regular topics
keywords	a list of keywords
model_settings	a list of model specific settings (details are in the online documentation)
priors	a list of priors of parameters
options	a list of options

- **seed**: A numeric value for random seed. If it is not provided, the package randomly selects a seed.

- **iterations**: An integer. Number of iterations. Default is 1500.
- **verbose**: If `TRUE`, it prints loglikelihood and perplexity. Default is `FALSE`.
- **llk_per**: An integer. If the value is `j` **keyATM** stores loglikelihood and perplexity every `j` iteration. Default value is 10 per iterations
- **use_weights**: If `TRUE` use weight. Default is `TRUE`.
- **weights_type**: There are four types of weights. Weights based on the information theory (`information-theory`) and inverse frequency (`inv-freq`) and normalized versions of them (`information-theory-normalized` and `inv-freq-normalized`). Default is `information-theory`.
- **prune**: If `TRUE` rume keywords that do not appear in the corpus. Default is `TRUE`.
- **store_theta**: If `TRUE` or `1`, it stores *theta* (document-topic distribution) for the iteration specified by thinning. Default is `FALSE` (same as `0`).
- **store_pi**: If `TRUE` or `1`, it stores *pi* (the probability of using keyword topic word distribution) for the iteration specified by thinning. Default is `FALSE` (same as `0`).
- **thinning**: An integer. If the value is `j` **keyATM** stores following parameters every `j` iteration. The default is `5`.
  - *theta*: For all models. If `store_theta` is `TRUE` document-level topic assignment is stored (sufficient statistics to calculate document-topic distributions `theta`).
  - *alpha*: For the base and dynamic models. In the base model alpha is shared across all documents whereas each state has different alpha in the dynamic model.
  - *lambda*: For the covariate model.
  - *R*: For the dynamic model. The state each document belongs to.
  - *P*: For the dynamic model. The state transition probability.
- **parallel_init**: Parallelize processes to speed up initialization. Default is `FALSE`. Note that even if you use the same `seed`, the initialization will become different between with and without parallelization.

`keep` a vector of the names of elements you want to keep in output

Value

A keyATM_output object containing:

keyword_k number of keyword topics

no_keyword_topics number of no-keyword topics

V number of terms (number of unique words)

N number of documents

model the name of the model

theta topic proportions for each document (document-topic distribution)

phi topic specific word generation probabilities (topic-word distribution)

topic_counts number of tokens assigned to each topic

word_counts number of times each word type appears

doc_lens length of each document in tokens

vocab words in the vocabulary (a vector of unique words)

priors priors

options options

keywords_raw specified keywords

model_fit perplexity and log-likelihood

pi estimated pi for the last iteration

values_iter values stored during iterations

kept_values outputs you specified to store in keep option

information information about the fitting

See Also

https://keyatm.github.io/keyATM/articles/pkgdown_files/Options.html

Examples

```
## Not run:
  library(keyATM)
  library(quanteda)
  data(keyATM_data_bills)
  bills_keywords <- keyATM_data_bills$keywords
  bills_dfm <- keyATM_data_bills$doc_dfm  # quanteda dfm object
  keyATM_docs <- keyATM_read(bills_dfm)
  # keyATM Base
  out <- keyATM(
               docs, model = "base", no_keyword_topics = 5, keywords = keywords_list
              )

  # keyATM Covariates
  out <- keyATM(
             docs, model = "covariates", no_keyword_topics = 5, keywords = keywords_list,
                model_settings(covariates_data = cov, covariates_formula = ~ .)
              )

  # keyATM Dynamic
  out <- keyATM(
               docs, model = "dynamic", no_keyword_topics = 5, keywords = keywords_list,
                model_settings(time_index = time_index_vec, num_states = 5)
              )

  # Visit our website for full examples: https://keyatm.github.io/keyATM/


## End(Not run)
```

keyATM_data_bills *Bills data*

Description

Bills data

Usage

```
keyATM_data_bills
```

Format

A list with following objects:

doc_dfm A quanteda dfm object of 140 documents. The text data is a part of the Congressional Bills scraped from https://www.congress.gov.

cov An integer vector which takes one if the Republican proposed the bill.

keywords A list of length 4 which contains keywords for four selected topics.

time_index An integer vector indicating the session number of each bill.

labels An integer vector indicating 40 labels.

labels_all An integer vector indicating all labels.

Source

https://www.congress.gov

keyATM_read *Read texts*

Description

Read texts and create a keyATM_docs object, which is a list of texts.

Usage

```
keyATM_read(texts, encoding = "UTF-8", check = TRUE)
```

Arguments

texts	input. keyATM takes dfm, data.frame, **tibble** tbl_df, or a vector of file paths.
encoding	character. Only used when texts is a vector of file paths. Default is "UTF-8".
check	logical. If TRUE, check whether there is nothing wrong with the structure of texts. Default is TRUE.

Value

a list whose elements are splitted texts. The length of the list equals to the number of documents.

Examples

```
## Not run:
 # Use quanteda dfm
 keyATM_docs <- keyATM_read(texts = quanteda_dfm)

 # Use data.frame or tibble (texts should be stored in a column named `text`)
 keyATM_docs <- keyATM_read(texts = data_frame_object)
 keyATM_docs <- keyATM_read(texts = tibble_object)

 # Use a vector that stores full paths to the text files
 files <- list.files(doc_folder, pattern = "*.txt", full.names = TRUE)
 keyATM_docs <- keyATM_read(texts = files)


## End(Not run)
```

---

plot.strata_doctopic *Plot document-topic distribution by strata (for covariate models)*

Description

Plot document-topic distribution by strata (for covariate models)

Usage

```
## S3 method for class 'strata_doctopic'
plot(x, topics = NULL, quantile_vec = c(0.05, 0.5, 0.95), ...)
```

Arguments

x	a strata_doctopic object (see by_strata_DocTopic())
topics	a vector or an integer. Indicate topics to visualize.
quantile_vec	a numeric. Quantiles to visualize
...	additional arguments not used

Value

ggplot2 object

plot_alpha *Show a diagnosis plot of alpha*

Description

Show a diagnosis plot of alpha

Usage

```
plot_alpha(x, start = 0, show_topic = NULL, scale = "fixed")
```

Arguments

x	the output from a keyATM model (see keyATM())
start	integer. The start of slice iteration. Default is 0.
show_topic	a vector to specify topic indexes to show. Default is NULL.
scale	character. Control the scale of y-axis (the parameter in facet_wrap()): 'free' adjusts y-axis for parameters. Default is "fixed".

Value

ggplot2 object

plot_modelfit *Show a diagnosis plot of log-likelihood and perplexity*

Description

Show a diagnosis plot of log-likelihood and perplexity

Usage

```
plot_modelfit(x, start = 1)
```

Arguments

x	the output from a keyATM model (see keyATM())
start	integer. The starting value of iteration to use in plot. Default is 1.

Value

ggplot2 object

plot_pi *Show a diagnosis plot of pi*

Description

Show a diagnosis plot of pi

Usage

```
plot_pi(x, show_topic = NULL, start = 0)
```

Arguments

x	the output from a keyATM model (see keyATM())
show_topic	an integer or a vector. Indicate topics to visualize. Default is NULL.
start	integer. The starting value of iteration to use in the plot. Default is 0.

Value

ggplot2 object

save.keyATM_output *Save a keyATM_output object*

Description

Save a keyATM_output object

Usage

```
save.keyATM_output(x, file = stop("'file' must be specified"))
```

Arguments

x	a keyATM_output object (see keyATM())
file	a character

save.keyATM_viz *Save a keyATM_viz object*

Description

Save a keyATM_viz object

Usage

```
save.keyATM_viz(x, file = stop("'file' must be specified"))
```

Arguments

x	a keyATM_viz object (see visualize_keywords())
file	a character

save_fig.keyATM_viz *Save a keyword plot*

Description

Save a keyword plot

Usage

```
save_fig.keyATM_viz(x, file = stop("'file' must be specified"))
```

Arguments

x	a keyATM_viz object (see visualize_keywords())
file	a character

top_docs *Show the top documents for each topic*

Description

Show the top documents for each topic

Usage

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

Arguments

x	the output from a keyATM model (see `keyATM_output()`)
n	How many documents to show. Default: 10

Value

An n x k table of the top n documents for each topic, each number is a document index

top_topics *Show the top topics for each document*

Description

Show the top topics for each document

Usage

```
top_topics(x, n = 2)
```

Arguments

x	the output from a keyATM model (see `keyATM()`)
n	integer. The number of topics to show. Default is 2.

Value

An n x k table of the top n topics in each document

top_words *Show the top words for each topic*

Description

If show_keyword is true then words in their seeded categories are suffixed with a check mark. Words from another seeded category are labeled with the name of that category.

Usage

```
top_words(x, n = 10, measure = c("probability", "lift"), show_keyword = TRUE)
```

Arguments

x	the output (see keyATM() and by_strata_TopicWord())
n	integer. The number terms to visualize. Default is NULL, which shows all terms.
measure	character. The way to sort the terms: 'probability' (default) or 'lift'.
show_keyword	logical. If TRUE, mark keywords. Default is TRUE.

Value

An n x k table of the top n words in each topic

visualize_keywords *Visualize keywords*

Description

Visualize the proportion of keywords in the documents.

Usage

```
visualize_keywords(docs, keywords, prune = TRUE, label_size = 3.2)
```

Arguments

docs	a keyATM_docs object, generated by keyATM_read() function
keywords	a list of keywords
prune	logical. If TRUE, prune keywords that do not appear in 'docs'. Default is TRUE.
label_size	the size of keyword labels in the output plot. Default is 3.2.

Value

A list containing

figure a ggplot2 object

values a tibble object that stores values

keywords a list of keywords that appear in documents

Examples

```
## Not run:
 # Prepare a keyATM_docs object
 keyATM_docs <- keyATM_read(input)

 # Keywords are in a list
 keywords <- list(
                  c("education", "child", "student"),  # Education
                  c("public", "health", "program"),  # Health
                )

 # Visualize keywords
 keyATM_viz <- visualize_keywords(keyATM_docs, keywords)

 # View a figure
 keyATM_viz
   # Or: `keyATM_viz$figure`

 # Save a figure
 save_fig(keyATM_viz, filename)


## End(Not run)
```

`weightedLDA` *Weighted LDA main function*

Description

Run weighted LDA models.

Usage

```
weightedLDA(
  docs,
  model,
  number_of_topics,
  model_settings = list(),
  priors = list(),
```

```
  options = list(),
  keep = c()
)
```

Arguments

`docs`	texts read via `keyATM_read()`
`model`	Weighted LDA model: "base", "covariates", and "dynamic"
`number_of_topics`	the number of regular topics
`model_settings`	a list of model specific settings (details are in the online documentation)
`priors`	a list of priors of parameters
`options`	a list of options (details are in the documentation of `keyATM()`)
`keep`	a vector of the names of elements you want to keep in output

Value

A keyATM_output object containing:

V number of terms (number of unique words)

N number of documents

model the name of the model

theta topic proportions for each document (document-topic distribution)

phi topic specific word generation probabilities (topic-word distribution)

topic_counts number of tokens assigned to each topic

word_counts number of times each word type appears

doc_lens length of each document in tokens

vocab words in the vocabulary (a vector of unique words)

priors priors

options options

keywords_raw `NULL` for LDA models

model_fit perplexity and log-likelihood

pi estimated pi for the last iteration (`NULL` for LDA models)

values_iter values stored during iterations

number_of_topics number of topics

kept_values outputs you specified to store in `keep` option

information information about the fitting

See Also

https://keyatm.github.io/keyATM/articles/pkgdown_files/Options.html

Examples

```
## Not run:
  # Weighted LDA
  out <- weightedLDA(
                     keyATM_docs, model = "base", number_of_topics = 5
                    )

  # Weighted LDA Covariates
  out <- weightedLDA(
                     keyATM_docs, model = "covariates", number_of_topics = 5,
                     model_settings(covariates_data = cov, covariates_formula = ~ .)
                    )

  # Weighted LDA Dynamic
  out <- weightedLDA(
                     keyATM_docs, model = "dynamic", number_of_topics = 5,
                     model_settings(time_index = time_index_vec, num_states = 5)
                    )

  # Visit our website for full examples: https://keyatm.github.io/keyATM/


## End(Not run)
```

Index