# R documentation

of
'/Users/shusei/Dropbox/Study/Project/ImaiText/keyATM/man/keyATM.Rd'
etc.

April 20, 2020

---

`keyATM` *keyATM main function*

---

## Description

Fit 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<br>• **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*: coefficients in 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 $\pi$ (the probability of using keyword topic word distribution) 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

```
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(keyATM_docs, model = "base", no_keyword_topics = 5,
              keywords = bills_keywords)

# keyATM Covariates
bills_cov <- as.data.frame(keyATM_data_bills$cov)
out <- keyATM(keyATM_docs, model = "covariates", no_keyword_topics = 5,
              keywords = bills_keywords,
              model_settings = list(covariates_data = bills_cov,
                                    covariates_formula = ~RepParty))

# keyATM Dynamic
bills_time_index <- keyATM_data_bills$time_index
bills_time_index <- as.integer(bills_time_index - 100)  # starts from 1, increment by 1
out <- keyATM(keyATM_docs, model = "dynamic", no_keyword_topics = 5,
              keywords = bills_keywords,
              model_settings = list(num_states = 5,
                                    time_index = bills_time_index))

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

---

`weightedLDA` *Weighted LDA main function*

---

## Description

Fit 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

```
library(keyATM)
library(quanteda)
data(keyATM_data_bills)
bills_dfm <- keyATM_data_bills$doc_dfm  # quanteda dfm object
keyATM_docs <- keyATM_read(bills_dfm)

# Weighted LDA
out <- weightedLDA(keyATM_docs, model = "base", number_of_topics = 5)

# Weighted LDA Covariates
bills_cov <- as.data.frame(keyATM_data_bills$cov)
out <- weightedLDA(keyATM_docs, model = "covariates", number_of_topics = 5,
                   model_settings = list(covariates_data = bills_cov,
                                         covariates_formula = ~ RepParty))

# Weighted LDA Dynamic
bills_time_index <- keyATM_data_bills$time_index
bills_time_index <- as.integer(bills_time_index - 100)  # starts from 1, increment by 1
out <- weightedLDA(keyATM_docs, model = "dynamic", number_of_topics = 5,
                   model_settings = list(num_states = 5,
                                         time_index = bills_time_index))

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

# Index