

# Package ‘LDAShiny’

January 20, 2025

**Title** User-Friendly Interface for Review of Scientific Literature

**Version** 0.9.3

**Maintainer** Javier De La Hoz Maestre <jdelahozmaestre@gmail.com>

**BugReports** <https://github.com/JavierDeLaHoz/LDAShiny/issues>

**Description** Contains the development of a tool that provides a web-based graphical user interface (GUI) to perform a review of the scientific literature under the Bayesian approach of Latent Dirichlet Allocation (LDA) and machine learning algorithms. The application methodology is framed by the well known procedures in topic modelling on how to clean and process data. Contains methods described by Blei, David M., Andrew Y. Ng, and Michael I. Jordan (2003) <<https://jmlr.org/papers/volume3/blei03a/blei03a.pdf>> "Allocation"; Thomas L. Griffiths and Mark Steyvers (2004) <[doi:10.1073/pnas.0307752101](https://doi.org/10.1073/pnas.0307752101)> ; Xiong Hui, et al (2019) <[doi:10.1016/j.cie.2019.06.010](https://doi.org/10.1016/j.cie.2019.06.010)>.

**License** GPL-3

**Imports** beepR, broom, chinese.misc, dplyr, DT (>= 0.15), highcharter, htmlwidgets, ldatuning, parallel, plotly, purrr, quanteda, shiny, shinyalert, shinyBS, shinycssloaders, shinydashboard, shinyjs, shinyWidgets, SnowballC, stringr, textmineR, tidyr, tidytext, tm, topicmodels

**Suggests** knitr, RColorBrewer, rmarkdown, Rmpfr, scales, magrittr

**VignetteBuilder** knitr

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Javier De La Hoz Maestre [cre, aut]

(<<https://orcid.org/0000-0001-7779-0803>>),

María José Fernández Gómez [aut]

(<<https://orcid.org/0000-0002-5530-6416>>),

Susana Mendez [aut] (<<https://orcid.org/0000-0001-9681-3169>>)

**Repository** CRAN

**Date/Publication** 2021-03-29 10:02:12 UTC

## Contents

crude . . . . .	2
removeSparseTerms . . . . .	3
runLDAShiny . . . . .	3
<b>Index</b>	<b>5</b>

---

crude	<i>20 Exemplary News Articles from the Reuters-21578 Data Set of Topic crude</i>
-------	--

---

## Description

This data set holds 20 news articles with additional meta information from the Reuters-21578 data set. All documents belong to the topic crude dealing with crude oil

## Usage

```
data("crude")
```

## Format

A VCorpus of 20 text documents. source Reuters-21578 Text Categorization Collection Distribution 1.0 XML format

## Value

Object class VCorpus

## References

Emms, Martin and Luz, Saturnino (2007). Machine Learning for Natural Language Processing. *European Summer School of Logic, Language and Information, course reader.*

## Examples

```
data("crude")
crude
```

---

removeSparseTerms	<i>removeSparseTerms Remove Sparse Terms from a Term-Document Matrix function original package tm</i>
-------------------	---

---

**Description**

removeSparseTerms Remove Sparse Terms from a Term-Document Matrix function original package tm

**Usage**

```
removeSparseTerms(x, sparse)
```

**Arguments**

x	A DocumentTermMatrix or a TermDocumentMatrix
sparse	A numeric for the maximal allowed sparsity in the range from bigger zero to smaller one.

**Value**

A term-document matrix where those terms from x are removed which have at least a sparse percentage of empty (i.e., terms occurring 0 times in a document) elements. I.e., the resulting matrix contains only terms with a sparse factor of less than sparse

**Examples**

```
data("crude")
require(tm)
tdm <- tm::TermDocumentMatrix(crude)
removeSparseTerms(tdm, 0.3)
```

---

runLDAShiny	<i>Shiny UI for LDAShiny package</i>
-------------	--------------------------------------

---

**Description**

Shiny UI for LDAShiny package

**Usage**

```
runLDAShiny(host = "127.0.0.1", port = NULL, launch.browser = TRUE)
```

**Arguments**

<code>host</code>	The IPv4 address that the application should listen on. Defaults to the <code>shiny.host</code> option, if set, or "127.0.0.1" if not.
<code>port</code>	is the TCP port that the application should listen on. If the port is not specified, and the <code>shiny.port</code> option is set (with <code>options(shiny.port = XX)</code> ), then that port will be used. Otherwise, use a random port.
<code>launch.browser</code>	If true, the system's default web browser will be launched automatically after the app is started. Defaults to true in interactive sessions only. This value of this parameter can also be a function to call with the application's URL.

**Value**

No return value, just start the GUI

**Examples**

```
if (interactive()) {  
  runLDAShiny()  
}
```

# Index

\* **datasets**

crude, [2](#)

crude, [2](#)

removeSparseTerms, [3](#)

runLDAShiny, [3](#)