Package ‘roclang’

May 26, 2023

**Title**  Functions for Diffusing Function Documentations into 'Roxygen' Comments

**Version**  0.2.2

**Maintainer**  Xiurui Zhu <zxr6@163.com>

**Description**  Efficient diffusing of content across function documentations. Sections, parameters or dot parameters are extracted from function documentations and turned into valid Rd character strings, which are ready to diffuse into the 'roxygen' comments of another function by inserting inline code.

**License**  MIT + file LICENSE

**Suggests**  covr, testthat (>= 3.0.0)

**Config/testthat/edition**  3

**Encoding**  UTF-8

**RoxygenNote**  7.2.0

**Depends**  R (>= 4.0.0)

**Imports**  dplyr (>= 1.0.2), tidyr (>= 1.1.2), purrr (>= 0.3.4), tibble (>= 3.0.4), stringr (>= 1.4.0), magrittr (>= 2.0.1), rlang (>= 0.4.10), roxygen2 (>= 7.1.1), methods (>= 4.0.0), utils (>= 4.0.0), rex (>= 1.2.0)

**URL**  https://github.com/zhuxr11/roclang

**BugReports**  https://github.com/zhuxr11/roclang/issues

**NeedsCompilation**  no

**Author**  Xiurui Zhu [aut, cre]

**Repository**  CRAN

**Date/Publication**  2023-05-26 12:10:02 UTC

### R topics documented:

<table>
<thead>
<tr>
<th>Package</th>
<th>Topic</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>roclang</td>
<td>roclang-package</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>extract_roc_text</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>roc_eval_text</td>
<td>5</td>
</tr>
</tbody>
</table>

**Index**  7
roclang-package

roclang: A package for diffusing function documentations into 'roxygen' comments

Description

The `roclang` package facilitates efficient diffusing of content across function documentations. Sections, parameters or dot parameters are extracted from function documentations and turned into valid Rd character strings, which are ready to diffuse into the 'roxygen’ comments of another function by inserting inline code.

Functions

- Text extraction and manipulation function: `extract_roc_text`.
- Rd evaluation and compilation function: `roc_eval_text`.

Note

Change log:

- 0.1.1 Xiurui Zhu - Initiate the document.

Author(s)

Xiurui Zhu

extract_roc_text

Extract a section, parameter or set of dot-parameters from a function documentation

Description

`extract_roc_text` cites sections or parameters from a function documentation in the syntax of `@inherit, @inheritSection, @inheritParams` or `@inheritDotParams` tag from `roxygen2` package. See details about how to use this function.

Usage

```r
extract_roc_text(
  fun,
  type = c("general", "section", "param", "dot_params"),
  select = NULL,
  capitalize = NA
)
```
Arguments

fun Function or character (of length 1L) indicating function name.

type Type of extraction. Please choose one from the following table according to the @tag you would otherwise use if you would like to inherit the section, parameter or set of dot-parameters as a whole:

<table>
<thead>
<tr>
<th>@tag you would use</th>
<th>type you should choose</th>
</tr>
</thead>
<tbody>
<tr>
<td>@inherit</td>
<td>&quot;general&quot;</td>
</tr>
<tr>
<td>@inheritSection</td>
<td>&quot;section&quot;</td>
</tr>
<tr>
<td>@inheritParams</td>
<td>&quot;param&quot;</td>
</tr>
<tr>
<td>@inheritDotParams</td>
<td>&quot;dot_params&quot;</td>
</tr>
</tbody>
</table>

select Selection of extraction based on type.

type = "general" Character (of length 1L) indicating the section to extract

type = "section" Character (of length 1L) indicating the section title to extract


type = "param" Character (of length 1L) indicating the name of parameter to extract


type = "dot_params" Character (of length 1L) or character vector to add or remove (with "-") parameters as @inheritDotParams; if character vector provided, the elements are concatenated with spaces just as @inheritDotParams syntax, e.g. "x y" to inherit two parameters, "-z" to remove a parameter or c("-x", "+y") to remove two parameters

capitalize Logical (of length 1L) indicating whether the first letter of the return should be capitalized. Default to capitalize = NA, in which case the first letter of the return is left as is.

Details

To diffuse the function output into roxygen2 comments, one may write the function documentation with inline code like this:

```
#' Diffusion of function documentation with inline code
#' @return Same as \code{\link[stats]{lm}}:
#' \r extract_roc_text(stats::lm, type = "general", select = "return")
my_fun <- function() {} 
```

or with code block like this:

```
#' Diffusion of function documentation with code block
#' 
#' @param lm_arg Named list of
#' \r
#' extract_roc_text(stats::lm, 
#' type = "dot_params",
```
#' select = c("-formula", "-data"),
#' capitalize = FALSE)

my_fun <- function(lm_arg) {}

Value

Character (of length 1L) as a valid Rd character string to diffuse into roxygen2 comments.

Note

Change log:

• 0.1.0 Xiurui Zhu - Initiate the function.
• 0.1.1 Xiurui Zhu - Change the default of capitalize from TRUE to NA.
• 0.1.1 Xiurui Zhu - Improve code security in evaluating the formal arguments of fun.
• 0.2.0 Xiurui Zhu - Make changes for roxygen2 > 7.1.2 while keeping compatibility.

Author(s)

Xiurui Zhu

Examples

# Inherit a standard section, and leave the first letter as is
cat(
  extract_roc_text(stats::lm,
    type = "general",
    select = "description",
    capitalize = NA)
)

# Inherit a self-defined section, and capitalize the first letter
cat(
  extract_roc_text(stats::lm,
    type = "section",
    select = "Using time series",
    capitalize = TRUE)
)

# Inherit a parameter, and diffuse it into text
cat(
  paste0(
    "Here is the `formula` argument of `stats::lm`, defined as: ",
    extract_roc_text(stats::lm,
      type = "param",
      select = "formula",
      capitalize = FALSE)
  )
)
# Inherit a set of dot params, and diffuse it into text

cat(
  paste0(
    "lm_arg" is a named list of ",
    extract_roc_text(stats::lm,
      type = "dot_params",
      select = c("-formula", 
      capitalize = FALSE)
    )
  )
)

---

roc_eval_text Generate Rd from text with evaluated inline code and code blocks

**Description**

roc_eval_text is an upgraded version of roc_proc_text that evaluates inline and block code before generating Rd.

**Usage**

roc_eval_text(roclet, input)

**Arguments**

- **roclet** Name of roclet to use for processing.
- **input** Source string

**Value**

List with names as fun_name.Rd, where each element is the RoxyTopic for the corresponding function, same as the return of roc_proc_text.

**Note**

Change log:

- 0.1.0 Xiurui Zhu - Initiate the function.

**Author(s)**

Xiurui Zhu
Examples

# Formulate a text version of a function with documentation
fun_text <- '  #' \code{iris} is a `r nrow(iris)`-row matrix.
  #' \code{iris} matrix has  
  #' ``(r results="hold")
  #' ncol(iris)
  #' ``
  #' columns.
  print_iris <- function() iris
  '

  # Parse the 'roxygen' comments to Rd documentation
  roc_eval_text(roxygen2::rd_roclet(), fun_text)[[1L]]
Index

extract_roc_text, 2, 2
roc_eval_text, 2, 5
roc_proc_text, 5
roclang (roclang-package), 2
roclang-package, 2
roxygen2, 2–4
RoxyTopic, 5