Package ‘formatBibtex’

March 4, 2025

Title Format BibTeX Entries and Files

Version 0.1.1

Description Format BibTeX entries and files in an opinionated way.

Depends R (>= 4.0)

Imports utils

Suggests bibtex

License GPL (>= 3)

Encoding UTF-8

URL https://wwenjie.org/formatbibtex,
https://github.com/wenjie2wang/formatBibtex

BugReports https://github.com/wenjie2wang/formatBibtex/issues

RoxygenNote 7.3.2

NeedsCompilation no

Author Wenjie Wang [aut, cre] (<https://orcid.org/0000-0003-0363-3180>)

Maintainer Wenjie Wang <wang@wwenjie.org>

Repository CRAN

Date/Publication 2025-03-04 03:30:02 UTC

Contents

format_bibtex_entry *Format BibTeX Entries in An Opinionated Way*

Description

This function tries to format the given BibTeX entries so that

- The “title” field is written in the title case with exceptations such as common prepositions with four or fewer letters. Special words such as “Bayesian” and “Markov” are protected by curly braces from case changing by BibTeX sytle.
- The “author” field follows a consistent fashion: “family name, given name”. A period will be added to single letter acronym.
- The “journal” field (if any) is written in the title case.
- The “pages” field should use “–” instead of “-” between pages.

Usage

```
format_bibtex_entry(
  entry,
  fields = c("title", "author", "journal", "pages"),
  protected_words = NULL,
  ...
)

format_bibtex_file(
  bibtex_file,
  output_file = bibtex_file,
  backup = (output_file == bibtex_file),
  dry_run = FALSE,
  ...
)
```

Arguments

`entry`	A `bibentry` object (created by `utils::bibentry`) representing BibTeX entries.
`fields`	A character vector specifying the fields to format. The available options are `"title"`, `"author"`, `"journal"`, and `"pages"`. Multiple choices can be specified.
`protected_words`	Optional words that needs protection by curly braces from cases changing by BibTeX style.
`...`	Other arguments passed to `format_bibtex_entry`.
`bibtex_file`	A character string presenting the BibTeX file that needs formatting.
`output_file`	A character string presenting the output BibTeX file. By default, the input BibTeX file will be overwritten with a backup file.

`backup` A logical value. If `TRUE`, a backup file will be created to check and tweak the formatting options.

`dry_run` A logical value. If `TRUE`, the formatted BibTeX entries will be returned without actually (over)writing a BibTeX file to the disk. The default value is `FALSE`.

Details

When `emacs` is available in the system, the function `format_bibtex_file()` will perform additional formatting with the help of the commands `bibtex-reformat` and `bibtex-sort-buffer`.

Value

A `bibtex` object.

Examples

```
library(formatBibtex)

## example BibTeX file that needs formatting
example_bib <- system.file("examples/example.bib", package = "formatBibtex")
print(readLines(example_bib), quote = FALSE)

## check the default words that need protection by curly braces
format_options$get("protected_words")
## append "SMEM" to the list
format_options$append("protected_words", c("SMEM"))

## needs the package bibtex
if (requireNamespace("bibtex", quietly = TRUE)) {
    ## example of format_bibtex_entry()
    bib <- bibtex::read.bib(example_bib)
    format_bibtex_entry(bib)
    ## example of format_bibtex_file()
    output_file <- tempfile(fileext = ".bib")
    format_bibtex_file(example_bib, output_file = output_file)
    print(readLines(output_file), quote = FALSE)
}
```

`format_options` *Package Options*

Description

A list of functions for getting, setting, and appending options for formatting. The available options are `"lowercase_words"`, `"punctuation"`, and `"protected_words"`. The default values can be retrieved by `formatBibtex_defaults$get()`.

Usage

```
format_options
```

Format

An object of class `list` of length 3.

Value

A list of functions.

`format_string` *Format a Character Vector*

Description

This function formats a character vector by following APA (American Psychological Assoication) style.

Usage

```
format_string(
  x,
  style = c("title", "sentence"),
  str2ws = "_",
  str4split = " |-",
  strict = TRUE,
  arguments = list(),
  protect_curly_braces = FALSE,
  lowercase_words = NULL,
  protected_words = NULL,
  punctuation = NULL,
  ...
)
```

Arguments

`x` A character vector that needs formatting.

`style` A character string for specifying case style. `"title"` for title case or `"sentence"` for sentence case.

`str2ws` Character expression that would be replaced with white space. By default, underscores will be replaced with white spaces.

`str4split` Character expression that would be used to split the string into individual words.

`strict` A logical value specifying whether to lower the cases for letters after the first letter. The default value is `TRUE`.

`arguments` An optional argument list for specifying more details when replacing specified characters with whitespaces by using `gsub`.

`protect_curly_braces` A logical vector of length one indicating if curly braces represent a block that needs protection. This argument is mainly intended to format BibTeX entries.

`lowercase_words`
: Some words that should be almost always be lowercase. The default values are `format_options$get("lowercase_words")`.

`protected_words`
: Some words that should be kept in original case and should be not be convert to lowercase or uppercase. The default values are `format_options$get("protected_words")`.

`punctuation`
: A character expression that should be considered as punctuation and should not be considered as a part of the protected words. The function would remove these punctuation before checking whether the words need protecting. The default values are `format_options$get("punctuation")`.

`...`
: Other arguments that are not used now.

Details

The available style options are title case and sentence case. The first word of the string and the first word after colon will be always capitalized. For title case (default), the function would capitalize each word with a few exceptions, such as short conjunctions and prepositions. For sentence case, the function would not capitalize a word unless it starts a string, come after a colon, or is a proper noun. We may specify some words that need protection from any kind of case conversion.

The function would also replace possible splitting characters, such as underscores with white spaces. Empty strings (`""` or `''`), `NA`, `character(0)`, and `NULL` are allowed and returned as they were.

Value

A character vector of the same length as the input.

Examples

```
library(formatBibtex)

## simple examples
foo <- c("iT IS_A_sIMPLe_ExamplE.", "let'S_do soMe_tesTs!")
format_string(foo, style = "title")
format_string(foo, style = "sentence")

## default words that would be protected from formatting
format_options$get("protected_words")

## e.g., protect ABCD and MCMC from being converted to lowercase
format_options$append("protected_words", c("ABCD", "MCMC"))

bar <- c("on_the_convergence properties of the ABCD_algorithm",
         "teSt: tHe cluster is Running MCMC!")
format_string(bar, style = "sentence")

## more tricky examples: protected words contain `str4split`
foo <- c("nineteenth- and twentieth-century writers",
         "well-differentiated cells with arXiv e-prints")
format_string(foo, str4split = "-| ",
              protected_words = c("arXiv", "e-prints"))
```

```
## trivial examples
format_string(NULL)
format_string(character(0))
format_string(character(3))
format_string(c(NA, "", "hello world!"))
```

Index