

# Package ‘litRiddle’

July 19, 2023

**Type** Package

**Title** Dataset and Tools to Research the Riddle of Literary Quality

**Version** 1.0.0

**Date** 2023-07-17

**Author** Maciej Eder [aut, cre],  
Joris van Zundert [aut],  
Karina van Dalen-Oskam [aut],  
Saskia Lensink [aut]

**Maintainer** Maciej Eder <maciejeder@gmail.com>

**URL** <https://literaryquality.huygens.knaw.nl/>

**Depends** R (>= 3.5.0)

**Imports** dplyr, ggplot2

**Suggests** stylo, knitr, rmarkdown

**Description** Dataset and functions to explore quality of literary novels. The package is a part of the Riddle of Literary Quality project, and it contains the data of a reader survey about fiction in Dutch, a description of the novels the readers rated, and the results of stylistic measurements of the novels. The package also contains functions to combine, analyze, and visualize these data. For more details, see: Eder M, van Zundert J, Lensink S, van Dalen-Oskam K (2022). Replicating The Riddle of Literary Quality: The litRiddle package for R. In *Digital Humanities 2022: Conference Abstracts*, 636-637.

**License** GPL (>= 3)

**Encoding** UTF-8

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2023-07-19 21:40:12 UTC

## R topics documented:

litRiddle-package . . . . .	2
books . . . . .	4
combine.all . . . . .	5
explain . . . . .	6
find.dataset . . . . .	7
frequencies . . . . .	8
get.columns . . . . .	9
make.table . . . . .	9
make.table2 . . . . .	11
motivations . . . . .	12
motivations.sentences . . . . .	13
motivations.text . . . . .	14
order.responses . . . . .	15
respondents . . . . .	16
reviews . . . . .	17
<b>Index</b>	<b>18</b>

---

litRiddle-package      *R Package to Research the Riddle of Literary Quality*

---

## Description

The package contains the data of a reader survey about fiction in Dutch, a description of the novels the readers rated, and the results of stylistic measurements of the novels. The package also contains functions to combine, analyze, and visualize these data.

We will be grateful if you cite the package in your publications. To get the updated citation information please type: `citation("litRiddle")`.

## Details

The package litRiddle presents data generated in the project The Riddle of Literary Quality (2012–2019) in which a team of digital humanists aimed to find out if books that readers considered to be highly literary have a different set of values for stylistic features than books the same readers did not consider to be very literary.

The package contains five data sets:

1. The reviews gathered from a hired representative panel of citizens of the Netherlands and in a large online survey called The National Reader Survey (2013). Type `help(reviews)` for details.
2. The motivations that reviewers give for a subset or all of their ratings are provided as plain text and as POS tagged data. Type `help(motivations)` for details.
3. Data about the reviewers: age, gender, zipcode, average number of books read per year etc. Type `help(respondents)` for details.

4. A list of the 401 books that the survey respondents evaluated with metadata such as author, title, publisher, gender of the author, and for translations the original language, etc., as well as a number of stylometric measurements such as the average sentence length etc. Type `help(books)` for details.
5. For each of the 401 books, the relative frequencies of 5000 most frequent words are provided (due to copyright issues the books themselves cannot be made available). Type `help(frequencies)` for details.

To learn more about the functions provided to analyze the above datasets, type the function `explain()` in your terminal.

### Author(s)

Maciej Eder, Joris van Zundert, Karina van Dalen-Oskam, Saskia Lensink

### References

Information in Dutch about the package can be found at [https://karinavdo.github.io/RaadseLliteratuur/02\\_07\\_data\\_en\\_R\\_package.html](https://karinavdo.github.io/RaadseLliteratuur/02_07_data_en_R_package.html)

Information in English at <https://github.com/karinavdo/LitRiddleData/blob/master/README.md>

Karina van Dalen-Oskam (2023). The Riddle of Literary Quality: A Computational Approach. Amsterdam University Press.

Karina van Dalen-Oskam (2021). Het raadsel literatuur. Is literaire kwaliteit meetbaar? Amsterdam University Press.

Maciej Eder, Saskia Lensink, Joris van Zundert, Karina van Dalen-Oskam (2022). Replicating The Riddle of Literary Quality: The litRiddle package for R, in: Digital Humanities 2022 Conference Abstracts. The University of Tokyo, Japan, 25–29 July 2022, p. 636-637 <https://dh2022.dhii.asia/dh2022bookofabsts.pdf>

Karina van Dalen-Oskam (2015). The Riddle of Literary Quality. Op zoek naar conventies van literariteit. "Vooy: tijdschrift voor letteren" 32(3): 25-33, <https://literaryquality.huygens.knaw.nl/?p=537#more-537>

Corina Koolen, Karina van Dalen-Oskam, Andreas van Cranenburgh, Erica Nagelhout (2020). Literary quality in the eye of the Dutch reader: The National Reader Survey. "Poetics" 79: 101439, doi: [10.1016/j.poetic.2020.101439](https://doi.org/10.1016/j.poetic.2020.101439)

More publications from the project: see [https://literaryquality.huygens.knaw.nl/?page\\_id=588](https://literaryquality.huygens.knaw.nl/?page_id=588)

### See Also

[books](#), [reviews](#), [respondents](#), [explain](#), [make.table](#)

---

books

*Measurements of 401 novels*

---

### Description

Measurements (including word count, number of sentences, number of paragraphs, average sentence length, etc.) of 401 novels in Dutch.

### Usage

```
data(books)
```

### Details

This is a dataframe containing numerical, ordinal and lexical data (as well as metadata) for 401 novels. To see which variables are provided, type `get.columns()`. To learn more about what the column names really mean, type `explain("books")`.

### Author(s)

Karina van Dalen-Oskam, Joris van Zundert

### Source

The dataset is a part of The Riddle of Literary Quality Project.

### See Also

[get.columns](#), [explain](#), [reviews](#), [respondents](#), [frequencies](#), [motivations](#)

### Examples

```
data(books)

print(books)
summary(books)
```

---

`combine.all`*Combine All Information of the Survey*

---

## Description

Function to combine all information of the survey, reviews, and books into one big dataframe. The user can specify whether or not they want to also load the `freqTable` with the frequency counts of the word n-grams of the books.

## Usage

```
combine.all(load.freq.table = FALSE)
```

## Arguments

`load.freq.table`

specify whether or not you want to add the `freqTable` with the frequency counts of the word n-grams of the books. Default is `FALSE`.

## Details

In order to identify (possible) correlations between particular reviews (e.g. the scores by the reviewers) with metadata about the reviewers themselves, it is usually required, or at least convenient, to combine two or more datasets into one large table.

## Value

A data frame combining the two (optionally three) datasets: books, respondents, and reviews.

## Author(s)

Saskia Lensink, Maciej Eder

## References

<https://literaryquality.huygens.knaw.nl/>

## See Also

[reviews](#), [respondents](#), [motivations](#), [books](#)

## Examples

```
# combine and load all data from the books, respondents and reviews into
# a new dataframe (tibble format)
combine.all(load.freq.table = FALSE)

# combine and load all data from the books, respondents and reviews into
# a new dataframe (tibble format), and additionally also load the frequency
```

```
# table of all word 1grams of the corpus used.  
combine.all(load.freq.table = TRUE)
```

---

explain

*Explain Variables*

---

### Description

Function that lists a short explanation of what the different column names refer to and what their levels consist of.

### Usage

```
explain(dataset = "")
```

### Arguments

dataset            specify whether or not you want to add the freqTable with the frequency counts of the word n-grams of the books. Default is FALSE.

### Details

In the current version, the option dataset = TRUE is not fully implemented.

### Value

A character vector being a description of the dataset.

### Author(s)

Saskia Lensink, Maciej Eder

### References

<https://literaryquality.huygens.knaw.nl/>

### See Also

[reviews](#), [respondents](#), [motivations](#), [books](#)

### Examples

```
explain("books")  
explain("reviews")  
explain("respondents")
```

---

find.dataset	<i>Find Dataset, Given a Column Name</i>
--------------	--

---

## Description

Return the name of the dataset where a column can be found.

## Usage

```
find.dataset(name = NULL)
```

## Arguments

name                    specify the name of the variable you want to find.

## Details

The function returns the name of the data table containing a given column name.

## Value

A character vector containing names of relevant datasets.

## Author(s)

Saskia Lensink, Maciej Eder

## References

<https://literaryquality.huygens.knaw.nl/>

## See Also

[reviews](#), [respondents](#), [motivations](#), [books](#)

## Examples

```
find.dataset("book.id")  
find.dataset("age.resp")
```

---

frequencies

*Word frequencies (5000 most frequent words) of 401 novels.*

---

### Description

Word frequencies (5000 most frequent words) of 401 novels in Dutch.

### Usage

```
data(frequencies)
```

### Details

This is a dataframe containing numerical values for word frequencies of the 5000 most frequent words (in a descending order of frequency) of 401 literary novels in Dutch. The table contains relative frequencies, meaning that the original word occurrences from a book were divided by the total number of words of the book in question. The measurements were obtained using the R package `stylo`, and were later rounded to the 5th digit. To learn more about the novels themselves, type `help(books)`.

### Author(s)

Karina van Dalen-Oskam, Maciej Eder

### Source

The dataset is a part of The Riddle of Literary Quality Project.

### See Also

[get.columns](#), [explain](#), [books](#), [reviews](#), [respondents](#), [motivations](#)

### Examples

```
data(frequencies)

print(frequencies)
summary(frequencies)
```



---

`get.columns`*Print Column Names*

---

### Description

The function creates a list of all the column names from all three datasets, i.d. [reviews](#), [respondents](#), [books](#).

### Usage

```
get.columns()
```

### Details

This simple function works best when combined with [explain](#), which provides a detailed description of particular variables. Type `help(explain)` for more details.

### Value

A list with three elements: `books`, `respondents`, and `reviews`, each containing the names of supported variables.

### Author(s)

Saskia Lensink, Maciej Eder

### References

<https://literaryquality.huygens.knaw.nl/>

### See Also

[reviews](#), [respondents](#), [books](#), [motivations](#), [explain](#)

---

`make.table`*Make Table and Plot*

---

### Description

A function to make a table of frequency counts for one variable, and to plot a histogram of the results.

**Usage**

```
make.table(table.of = NULL,  
           plot = TRUE,  
           xlab = table.of,  
           ylab = "count",  
           title = table.of,  
           barcolor = "grey",  
           barfill = "darkgrey")
```

**Arguments**

table.of	which variable will be chosen? If not sure what variables are there, try typing <code>get.columns()</code> first.
plot	do you want a plot to be plotted? Default: TRUE.
xlab	name of the X axis
ylab	name of the Y axis
title	title of the plot
barcolor	outline color of the content
barfill	color used to fill the bars

**Details**

A basic way to show the distribution of an indicated variable from the `litRiddle` package. It provides the values, but also a simple histogram.

**Value**

A character vector containing one chosen variable, optionally followed by a plot.

**Author(s)**

Saskia Lensink, Maciej Eder

**References**

<https://literaryquality.huygens.knaw.nl/>

**See Also**

[make.table2](#), [get.columns](#)

**Examples**

```
make.table(table.of = "age.resp")  
  
make.table(table.of = "age.resp", xlab = "age respondent",  
           ylab = "number of people", title = "Distribution of respondent age",  
           barcolor = "red", barfill = "white")
```

---

`make.table2`*Make Table of Two Variables and Plot*

---

**Description**

A function to make a table of frequency counts for two variables, and to plot a histogram of the results.

**Usage**

```
make.table2(table.of = NULL,  
            split = NULL,  
            plot = TRUE,  
            xlab = table.of,  
            ylab = "counts",  
            title = table.of,  
            barcolor = "grey",  
            barfill = "darkgrey")
```

**Arguments**

<code>table.of</code>	which variable will be chosen? If not sure what variables are there, try typing <code>get.columns()</code> first.
<code>split</code>	the variable that will be used to split the data: see the Examples section below for, well, some examples.
<code>plot</code>	do you want a plot to be plotted? Default: TRUE.
<code>xlab</code>	name of the X axis
<code>ylab</code>	name of the Y axis
<code>title</code>	title of the plot
<code>barcolor</code>	outline color of the content
<code>barfill</code>	color used to fill the bars

**Details**

Unlike `make.table`, this function provides a comparison of two variables at a time, or to be more precise: a distribution of an indicated variable when subdivided into two or more groups. The function provides the values themselves, but also a final histogram.

**Value**

A character vector containing one chosen variable, optionally followed by a plot.

**Author(s)**

Saskia Lensink, Maciej Eder

**References**

<https://literaryquality.huygens.knaw.nl/>

**See Also**

[make.table](#), [get.columns](#)

**Examples**

```
make.table2(table.of = "age.resp", split = "gender.resp")
make.table2(table.of = "literariness.read", split = "gender.author")

# Note that you can only provide an argument to the 'split' variable
# that has less than 31 unique values, to avoid uninterpretable outputs:
make.table2(table.of = "age.resp", split = "zipcode")

# You can also adjust the x label, y label, title, and colors.
make.table2(table.of = "age.resp", split = "gender.resp",
  xlab = "age respondent", ylab = "number of people",
  barcolor = "purple", barfill = "yellow")
make.table2(table.of = "literariness.read", split = "gender.author",
  xlab = "Overall literariness scores", ylab = "number of people",
  barcolor = "black", barfill = "darkred")
```

---

motivations

*Reviewers' motivations for their scores (if given)*

---

**Description**

Reviewers' motivations for their scores (if provided by the respondents) from the survey called The National Reader Survey (2013).

**Usage**

```
data(motivations)
```

**Details**

This is a dataframe containing that lists all tokens from all motivations together with lemma and POS tag information. To see which variables are provided, type `get.columns()`. To learn more about what the column names really mean, type `explain("motivations")`.

**Author(s)**

Karina van Dalen-Oskam, Joris van Zundert

**Source**

The dataset is a part of The Riddle of Literary Quality Project.

**See Also**

[get.columns](#), [explain](#), [books](#), [frequencies](#), [respondents](#), [reviews](#)

**Examples**

```
data(motivations)

head(motivations, n = 30)
summary(motivations)
```

---

`motivations.sentences` *Motivations Sentences*

---

**Description**

Convenience function that produces a 'view' of the token table `motivations` with one (plain text) sentence of each motivation per row, listening `motivation.id`, `book.id`, `respondent.id`, `sentence.id`, and `sentence`.

**Usage**

```
motivations.sentences()
```

**Arguments**

None

**Value**

A data table containing all sentences of all given motivations and IDs related to respondents and books.

**Author(s)**

Joris van Zundert, Saskia Lensink, Maciej Eder

**References**

<https://literaryquality.huygens.knaw.nl/>

**See Also**

[motivations.text](#), [reviews](#), [respondents](#), [books](#)

**Examples**

```
# to create a data frame with one sentence per motivation per row for all motivations:
mots <- motivations.sentences()
head( mots, n=10 )
```

---

motivations.text	<i>Motivations Text</i>
------------------	-------------------------

---

**Description**

Convenience function that produces a 'view' of the token table motivations with the full text of a motivation for each motivation, listening motivation.id, book.id, respondent.id, and text.

**Usage**

```
motivations.text()
```

**Arguments**

None

**Value**

A data table containing motivations and IDs related to respondents and books.

**Author(s)**

Joris van Zundert, Saskia Lensink, Maciej Eder

**References**

<https://literaryquality.huygens.knaw.nl/>

**See Also**

[motivations.sentences](#), [reviews](#), [respondents](#), [books](#)

**Examples**

```
# to create a data frame with the full (plain) text of all motivations:
mots <- motivations.text()
head( mots, n=10 )
```

---

order.responses	<i>Order Responses</i>
-----------------	------------------------

---

### Description

Function that transforms the survey responses into ordered factors. Levels `quality.read` and `quality.notread`: "very bad", "bad", "a bit bad", "neutral", "a bit good", "good", "very good", "NA". Levels `literariness.read` and `literariness.notread`: "absolutely not literary", "non-literary", "not very literary", "between literary and non-literary", "a bit literary", "literary", "very literary", "NA". Levels `statements 4/12`: "completely disagree", "disagree", "neutral", "agree", "completely agree", "NA".

### Usage

```
order.responses(bookratings.or.readingbehavior = NULL)
```

### Arguments

`bookratings.or.readingbehavior`

Use either "bookratings" or "readingbehavior" to specify which of the survey questions needs to be changed into ordered factors.

### Value

A data table containing relevant variables.

### Author(s)

Saskia Lensink, Maciej Eder

### References

<https://literaryquality.huylgens.knaw.nl/>

### See Also

[reviews](#), [respondents](#), [motivations](#), [books](#)

### Examples

```
# to create a data frame with ordered factor levels of the questions
# on reading behavior:
dat.reviews = order.responses("readingbehavior")
str(dat.reviews)

# to create a data frame with ordered factor levels of the book ratings:
dat.ratings = order.responses("bookratings")
str(dat.ratings)
```

---

respondents

*Respondents' Answers*

---

### **Description**

The information about the reviewers that participated in the survey called The National Reader Survey (2013).

### **Usage**

```
data(respondents)
```

### **Details**

This is a dataframe containing numerical, ordinal and textual data about the 13541 reviewers that scored 401 novels. To see which variables are provided, type `get.columns()`. To learn more about what the column names really mean, type `explain("respondents")`.

### **Author(s)**

Karina van Dalen-Oskam, Joris van Zundert

### **Source**

The dataset is a part of The Riddle of Literary Quality Project.

### **See Also**

[get.columns](#), [explain](#), [books](#), [reviews](#), [frequencies](#), [motivations](#)

### **Examples**

```
data(respondents)

print(respondents)
summary(respondents)
```



---

reviews

*Reviewers' scores*

---

**Description**

Reviewers' scores from the survey called The National Reader Survey (2013).

**Usage**

```
data(reviews)
```

**Details**

This is a dataframe containing numerical, ordinal and textual data for thousands of individual reviews (and the reviewers' scores) for 401 novels. To see which variables are provided, type `get.columns()`. To learn more about what the column names really mean, type `explain("reviews")`.

**Author(s)**

Karina van Dalen-Oskam, Joris van Zundert

**Source**

The dataset is a part of The Riddle of Literary Quality Project.

**See Also**

[get.columns](#), [explain](#), [books](#), [frequencies](#), [respondents](#), [motivations](#)

**Examples**

```
data(reviews)
```

```
print(reviews)
summary(reviews)
```

# Index

## \* datasets

- books, 4
- frequencies, 8
- motivations, 12
- respondents, 16
- reviews, 17

books, 3, 4, 5–9, 13–17

combine.all, 5

explain, 3, 4, 6, 8, 9, 13, 16, 17

find.dataset, 7

frequencies, 4, 8, 13, 16, 17

get.columns, 4, 8, 9, 10, 12, 13, 16, 17

litRiddle (litRiddle-package), 2

litRiddle-package, 2

make.table, 3, 9, 11, 12

make.table2, 10, 11

motivations, 4–9, 12, 15–17

motivations.sentences, 13, 14

motivations.text, 13, 14

order.responses, 15

respondents, 3–9, 13–15, 16, 17

reviews, 3–9, 13–16, 17