

Package ‘SemNetDictionaries’

July 21, 2025

Title Dictionaries for the 'SemNetCleaner' Package

Version 0.2.1

Date 2025-05-08

Maintainer Alexander P. Christensen <alexpaulchristensen@gmail.com>

Description Implements dictionaries that can be used in the 'SemNetCleaner' package. Also includes several functions aimed at facilitating the text cleaning analysis in the 'SemNetCleaner' package. This package is designed to integrate and update word lists and dictionaries based on each user's individual needs by allowing users to store and save their own dictionaries. Dictionaries can be added to the 'SemNetDictionaries' package by submitting user-defined dictionaries to <<https://github.com/AlexChristensen/SemNetDictionaries>>.

Depends R (>= 3.6.0)

License GPL (>= 3.0)

Suggests easycsv, ggplot2, gridExtra, htmlTable, knitr, markdown, patchwork, rmarkdown, shiny, shinyalert, shinyjs

URL <https://github.com/AlexChristensen/SemNetDictionaries>

BugReports <https://github.com/AlexChristensen/SemNetDictionaries/issues>

NeedsCompilation no

Encoding UTF-8

LazyData true

VignetteBuilder knitr

RoxygenNote 7.3.2

Author Alexander P. Christensen [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-9798-7037>>)

Repository CRAN

Date/Publication 2025-05-08 15:00:02 UTC

Contents

SemNetDictionaries-package	2
animals.dictionary	3
animals.moniker	3
append.dictionary	4
brit2us	6
coca.dictionary	6
coca.moniker	7
cocaspell.dictionary	7
cocaspell.moniker	8
dictionaries	8
find.dictionaries	9
fruits.dictionary	10
fruits.moniker	11
general.dictionary	11
good.dictionary	12
good.moniker	12
hot.dictionary	13
hot.moniker	13
hunspell.dictionary	14
jobs.dictionary	14
jobs.moniker	15
load.dictionaries	15
load.monikers	16
most_common.dictionary	17
ShinyWoRdle	18
stop_words.dictionary	18
vegetables.dictionary	19
vegetables.moniker	19
Index	20

SemNetDictionaries-package
<i>SemNetDictionaries-package</i>

Description

Implements dictionaries that can be used in the [SemNetCleaner-package](#). Also includes several functions aimed at facilitating the text cleaning analysis in the [SemNetCleaner-package](#). This package is designed to integrate and update word lists and dictionaries based on each user’s individual needs by allowing users to store and save their own dictionaries. Dictionaries can be added to the [SemNetDictionaries](#) package by submitting user-defined dictionaries to <https://github.com/AlexChristensen/SemNetDictionaries>

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

See Also

Useful links:

- <https://github.com/AlexChristensen/SemNetDictionaries>
- Report bugs at <https://github.com/AlexChristensen/SemNetDictionaries/issues>

animals.dictionary	<i>Animals Dictionary</i>
--------------------	---------------------------

Description

A database of possible animals responses ($n = 1211$)

Usage

```
data(animals.dictionary)
```

Format

animals.dictionary (vector, length = 1211)

Details

To add additional animals to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("animals.dictionary")
```

animals.moniker	<i>Animals Moniker</i>
-----------------	------------------------

Description

A database of possible animals monikers and common spelling errors

Usage

```
data(animals.moniker)
```

Format

animals.moniker (list, length = 236)

Details

To add additional animals monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen>

Examples

```
data("animals.moniker")
```

append.dictionary	<i>Appendix Dictionary</i>
-------------------	----------------------------

Description

A function designed to create post-hoc dictionaries in the [SemNetDictionaries](https://github.com/AlexChristensen/SemNetDictionaries) package. This allows for new semantic categories or word lists to be saved for future use (i.e., your own personal dictionary). Dictionaries created using this function can either be saved as an R object to your global environment or as a .rds file on your current computer. Open-source community-derived dictionaries can be uploaded to and downloaded from <https://github.com/AlexChristensen/SemNetDictionaries>

Usage

```
append.dictionary(
  ...,
  dictionary.name = "appendix",
  save.location = c("envir", "wd", "choose", "path"),
  path = NULL,
  textcleaner = FALSE,
  package = FALSE
)
```

Arguments

...	Character vector. A vector of words to create or add to a dictionary
dictionary.name	Character. Name of dictionary to create or add words to. Defaults to "appendix". Input a name to create or add to an existing dictionary. This function will automatically name files with the "*.dictionary.rds" suffix
save.location	Character. A choice for where to store appendix dictionary. Defaults to "envir". • "envir": Returns dictionary as a vector object to R's global environment • "wd": Saves dictionary to working directory. Useful for storing dictionaries alongside projects • "choose": User chooses a directory for more permanent storage. This will allow you to use this dictionary in the future • "path": User specifies a path to a directory if it is already known. This will allow direct updates to the directory and bypass the prompts in the save/update menus. This will also allow you to use this dictionary in the future
path	Character. A path to an existing directory. Only necessary for save.location = "path"

textcleaner	Boolean. Argument for skipping asking to save the dictionary twice. Defaults to FALSE. If TRUE, then asking to save the dictionary will be skipped.
package	Boolean. Argument not meant for user use. Allows me to update the package's dictionaries efficiently

Details

Appendix dictionaries are useful for storing spelling definitions that are not available in the [SemNetDictionaries](#) package. This function enables the storage of personalized dictionaries, which can be used in combination with other dictionaries to facilitate the cleaning of text data.

Dictionaries are either stored in R's global environment, where they will be deleted once R is closed (unless you save them), or in a directory you choose. A menu will pop-up asking whether you would like to save or update your dictionary. You have two options:

- Yes (or 1): Gives this function permission to save (or update) your dictionary to a chosen directory. If `save.location = "envir"`, your file will be deleted after closing R
- No (or 2): Does NOT give this function permission to save your dictionary to your computer. `save.location = "envir"` will always return your dictionary as a vector object to R's global environment

To save your dictionary file, you can either:

- Manually save: Use [saveRDS](#) and save using the `"*.dictionary"` suffix
- `save.location = "choose"`: A file explorer menu will pop-up and a directory can be manually selected
- `save.location = "path"`: The file will automatically be saved to the directory you provide

Note that `save.location = "choose"` and `save.location = "path"` will automatically update your dictionary if there is a file with the same name enter into the `dictionary.name` argument.

To find where your dictionaries are stored, use the [find.dictionaries](#) function. These dictionaries are only stored on your private computer and must either be publicly shared or transferred to other computers in order to use them elsewhere. If you would like to share a dictionary for others to use, then please submit a pull request or post an issue with your dictionary on my GitHub: [AlexChristensen/SemNetDictionaries](#).

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

See Also

[find.dictionaries](#) to find where dictionaries are stored, [dictionaries](#) to identify dictionaries in [SemNetDictionaries](#)

Examples

```
# Create a dictionary
new.dictionary <- append.dictionary(c("words", "are", "fun"), save.location = "envir")
```

brit2us	<i>British-US English Conversions</i>
---------	---------------------------------------

Description

A database to convert between British and US spellings ($n = 780$)

Usage

```
data(brit2us)
```

Format

```
brit2us (list, length = 780)
```

Examples

```
data("brit2us")
```

coca.dictionary	<i>Corpus of Contemporary American English Dictionary</i>
-----------------	---

Description

A general dictionary of over 80,000 words from the Corpus of Contemporary American English derived from <https://www.wordfrequency.info/samples.asp>.

Usage

```
data(coca.dictionary)
```

Format

```
coca.dictionary (vector, length = 80381)
```

Details

To add additional words to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("coca.dictionary")
```

`coca.moniker`*Corpus of Contemporary American English Moniker*

Description

A database of word forms for the Corpus of Contemporary American English dictionary

Usage

```
data(coca.moniker)
```

Format

```
coca.moniker (list, length = 20267)
```

Details

To add additional COCA monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen>

Examples

```
data("coca.moniker")
```

`cocaspell.dictionary`*Corpus of Contemporary American English and Hunspell Combined Dictionary*

Description

A general dictionary of over 109,000 words from the Corpus of Contemporary American English dictionary ([coca.dictionary](#)) and Hunspell dictionary ([hunspell.dictionary](#)).

Usage

```
data(cocaspell.dictionary)
```

Format

```
cocaspell.dictionary (vector, length = 109169)
```

Details

To add additional words to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("cocaspell.dictionary")
```

cocaspell.moniker	<i>Corpus of Contemporary American English and Hunspell Moniker</i>
-------------------	---

Description

A database of word forms for the Corpus of Contemporary American English and Hunspell dictionaries

Usage

```
data(cocaspell.moniker)
```

Format

```
cocaspell.moniker (list, length = 29610)
```

Details

To add additional COCA and Hunspell monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen/SemNetDictionaries>

Examples

```
data("cocaspell.moniker")
```

dictionaries	<i>List Names of Dictionaries in 'SemNetDictionaries'</i>
--------------	---

Description

A wrapper function to identify all dictionaries included in [SemNetDictionaries](#)

Usage

```
dictionaries(quiet)
```

Arguments

quiet	Boolean. Determines whether the return should be quiet (does not print dictionaries). Defaults to FALSE
-------	---

Value

Returns the names of dictionaries in [SemNetDictionaries](#)

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

See Also

[find.dictionaries](#) to find where dictionaries are stored, [append.dictionary](#) to create a new dictionary

Examples

```
# List names of dictionaries in 'SemNetDictionaries'
dictionaries()
```

find.dictionaries	<i>Finds Names and Locations of Appendix Dictionaries</i>
-------------------	---

Description

A wrapper function to identify the save location of appendix dictionaries from [append.dictionary](#)

Usage

```
find.dictionaries(..., add.path = NULL)
```

Arguments

...	Vector. Appendix dictionary files names (if they are known). If left empty, the function will search across all files for files in folders on your desktop that end in *.dictionary.rds. This search takes a few seconds to complete (see examples for your computer's exact timing)
add.path	Character. Path to additional dictionaries to be found. DOES NOT search recursively (through all folders in path) to avoid time intensive search. Set to "choose" to open an interactive directory explorer

Value

names	Returns the names of the appendix dictionary file(s) found on your computer
files	Returns the dictionary file(s) that are stored in each given path. If there is no output (e.g., <code>character(0)</code>), then no appendix dictionary file exists (one can be created using the append.dictionary function)

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

See Also

[append.dictionary](#) to create a new dictionary, [dictionaries](#) to identify dictionaries in [SemNetDictionaries](#), and [load.dictionaries](#) to load multiple dictionaries

Examples

```
# Make a dictionary
example.dictionary <- append.dictionary(c("words", "are", "fun"), save.location = "envir")

# Dictionary can now be found
find.dictionaries("example")

# No appendix dictionaries found
find.dictionaries()

# For your computer's timing to complete search
t0 <- Sys.time()
find.dictionaries()
Sys.time() - t0
```

fruits.dictionary	<i>Fruits Dictionary</i>
-------------------	--------------------------

Description

A database of possible fruits responses ($n = 488$)

Usage

```
data(fruits.dictionary)
```

Format

fruits.dictionary (vector, length = 488)

Details

To add additional fruits to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("fruits.dictionary")
```

fruits.moniker	<i>Fruits Moniker</i>
----------------	-----------------------

Description

A database of possible fruits monikers and common spelling errors

Usage

```
data(fruits.moniker)
```

Format

```
fruits.moniker (list, length = 39)
```

Details

To add additional fruits monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen/S>

Examples

```
data("fruits.moniker")
```

general.dictionary	<i>General Dictionary</i>
--------------------	---------------------------

Description

A general dictionary of over 370,000 words ($n = 370,103$) derived from <https://github.com/dwyl/english-words>. All punctuation have been removed.

Usage

```
data(general.dictionary)
```

Format

```
general.dictionary (vector, length = 370103)
```

Details

To add additional words to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("general.dictionary")
```

good.dictionary	<i>'Good' Synonyms Dictionary</i>
-----------------	-----------------------------------

Description

A database of possible good synonym responses ($n = 284$) To add additional good synonyms to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Usage

```
data(good.dictionary)
```

Format

```
good.dictionary (vector, length = 284)
```

Examples

```
data("good.dictionary")
```

good.moniker	<i>'Good' Moniker</i>
--------------	-----------------------

Description

A database of possible good monikers and common spelling errors

Usage

```
data(good.moniker)
```

Format

```
good.moniker (list, length = 4)
```

Details

To add additional good monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen/S>

Examples

```
data("good.moniker")
```

hot.dictionary	<i>'Hot' Synonyms Dictionary</i>
----------------	----------------------------------

Description

A database of possible hot synonym responses ($n = 281$) To add additional hot synonyms to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Usage

```
data(hot.dictionary)
```

Format

hot.dictionary (vector, length = 281)

Examples

```
data("hot.dictionary")
```

hot.moniker	<i>Hot Moniker</i>
-------------	--------------------

Description

A database of possible hot monikers and common spelling errors

Usage

```
data(hot.moniker)
```

Format

hot.moniker (list, length = 15)

Details

To add additional hot monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen/Ser>

Examples

```
data("hot.moniker")
```

hunspell.dictionary	<i>hunspell Dictionary</i>
---------------------	----------------------------

Description

A general dictionary of over 62,000 words from the hunspell dictionary derived from <http://wordlist.aspell.net/dicts/>.

Usage

```
data(hunspell.dictionary)
```

Format

hunspell.dictionary (vector, length = 62893)

Details

To add additional words to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("hunspell.dictionary")
```

jobs.dictionary	<i>Jobs Dictionary</i>
-----------------	------------------------

Description

A database of possible jobs and related words ($n = 1471$)

Usage

```
data(jobs.dictionary)
```

Format

jobs.dictionary (vector, length = 1471)

Details

To add additional jobs to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("jobs.dictionary")
```

jobs.moniker

Jobs Moniker

Description

A database of possible jobs monikers and common spelling errors

Usage

```
data(jobs.moniker)
```

Format

```
jobs.moniker (list, length = 117)
```

Details

To add additional jobs monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen/SemNetCleaner>

Examples

```
data("jobs.moniker")
```

load.dictionaries

Load Dictionaries

Description

A wrapper function to load dictionaries into the 'SemNetCleaner' package. Searches for dictionaries in R's global environment, the [SemNetDictionaries](#) package, and on your computer. Outputs a unique word list that is combined from all dictionaries entered in the dictionary argument

Usage

```
load.dictionaries(..., add.path = NULL)
```

Arguments

...	Character. Dictionaries to load Dictionaries in your global environment MUST be objects called "*.dictionary" (see examples)
add.path	Character. Path to additional dictionaries to be found. DOES NOT search recursively (through all folders in path) to avoid time intensive search. Set to "choose" to open an interactive directory explorer dictionaries will identify dictionaries in the SemNetDictionaries package find.dictionaries will identify dictionaries on your computer

Value

Returns a vector of unique words that have been combined and alphabetized from the specified dictionaries

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
# Find dictionaries to load
dictionaries()

# Load "animals" dictionary
load.dictionaries("animals")

# Create a dictionary
new.dictionary <- append.dictionary("words", "are", "fun")

# Load created dictionary
load.dictionaries("new")

# Load animals and new dictionary
load.dictionaries("animals", "new")

# Single letter dictionary
load.dictionaries("d")

# Multiple letters dictionary
load.dictionaries("a", "d")

# Category and letters dictionary
load.dictionaries("animals", "a")
```

load.monikers

Load Monikers

Description

A wrapper function to load monikers into the 'SemNetCleaner' package. Searches for monikers in R's [SemNetDictionaries](#) package. Outputs a unique word list that is combined from all dictionaries entered in the moniker argument

Usage

```
load.monikers(moniker, vector = TRUE)
```


Arguments

moniker	Character vector. monikers to load (must be a dictionary in dictionaries)
vector	Boolean. Should output be a vector? If FALSE, then output is a list. Defaults to TRUE

Value

Returns a vector of unique words that have been combined and alphabetized from the specified monikers

Author(s)

Alexander Christensen <alexpaulchristensen@gmail.com>

Examples

```
#find dictionaries to load
dictionaries()

#load "animals" monikers
load.monikers("animals")
```

```
most_common.dictionary
```

Most Common Dictionary

Description

A general dictionary of 10,000 of the most common U.S. English words derived from <https://github.com/first20hours/google-10000-english>.

Usage

```
data(most_common.dictionary)
```

Format

```
most_common.dictionary (vector, length = 9329)
```

Details

To add additional words to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("most_common.dictionary")
```

ShinyWoRdle

Shiny App to Play WoRdle

Description

An interactive Shiny application for playing WoRdle

Usage

```
ShinyWoRdle()
```

Examples

```
if(interactive())  
{ShinyWoRdle()}
```

stop_words.dictionary *Stop Words Dictionary*

Description

A selection of stop words that can be removed from semantic responses ($n = 56$)

Usage

```
data(stop_words.dictionary)
```

Format

```
stop_words.dictionary (vector, length = 56)
```

Details

To add additional animals to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("stop_words.dictionary")
```

vegetables.dictionary	<i>Vegetables Dictionary</i>
-----------------------	------------------------------

Description

A database of possible vegetables responses ($n = 284$)

Usage

```
data(vegetables.dictionary)
```

Format

vegetables.dictionary (vector, length = 284)

Details

To add additional vegetables to the dictionary, please make an appendix dictionary ([append.dictionary](#))

Examples

```
data("vegetables.dictionary")
```

vegetables.moniker	<i>Vegetables Moniker</i>
--------------------	---------------------------

Description

A database of possible vegetables monikers and common spelling errors

Usage

```
data(vegetables.moniker)
```

Format

vegetables.moniker (list, length = 35)

Details

To add additional vegetables monikers to the database, please submit a pull request or issue to <https://github.com/AlexChristensen/SemNetDictionaries>

Examples

```
data("vegetables.moniker")
```

Index

* datasets

- animals.dictionary, 3
- animals.moniker, 3
- brit2us, 6
- coca.dictionary, 6
- coca.moniker, 7
- cocaspell.dictionary, 7
- cocaspell.moniker, 8
- fruits.dictionary, 10
- fruits.moniker, 11
- general.dictionary, 11
- good.dictionary, 12
- good.moniker, 12
- hot.dictionary, 13
- hot.moniker, 13
- hunspell.dictionary, 14
- jobs.dictionary, 14
- jobs.moniker, 15
- most_common.dictionary, 17
- stop_words.dictionary, 18
- vegetables.dictionary, 19
- vegetables.moniker, 19

animals.dictionary, 3

animals.moniker, 3

append.dictionary, 3, 4, 6, 7, 9–14, 17–19

brit2us, 6

coca.dictionary, 6, 7

coca.moniker, 7

cocaspell.dictionary, 7

cocaspell.moniker, 8

dictionaries, 5, 8, 10, 15, 17

find.dictionaries, 5, 9, 9, 15

fruits.dictionary, 10

fruits.moniker, 11

general.dictionary, 11

good.dictionary, 12

good.moniker, 12

hot.dictionary, 13

hot.moniker, 13

hunspell.dictionary, 7, 14

jobs.dictionary, 14

jobs.moniker, 15

load.dictionaries, 10, 15

load.monikers, 16

most_common.dictionary, 17

saveRDS, 5

SemNetDictionaries, 2, 4, 5, 8, 10, 15, 16

SemNetDictionaries
(SemNetDictionaries-package), 2

SemNetDictionaries-package, 2

ShinyWoRdle, 18

stop_words.dictionary, 18

vegetables.dictionary, 19

vegetables.moniker, 19