Package 'SimilaR'

June 21, 2018

Version 1.0.2

Date 2018-06-21

Title R Source Code Similarity Evaluation

Description An Implementation of a novel method to determine similarity of R functions based on program dependence graphs, see Bartoszuk, Gagolewski (2017) <doi:10.1109/FUZZ-IEEE.2017.8015582>. Possible use cases include plagiarism detection among students' homework assignments.

URL http://similar.rexamine.com/

BugReports https://github.com/bartoszukm/SimilaR/issues

Type Package

Depends R (>= 3.1.0), stringi

License GPL (>= 3)

Imports Rcpp (>= 0.12.0)

Suggests testthat

LinkingTo Rcpp (>= 0.12.0), BH

SystemRequirements C++11

RoxygenNote 6.0.1

NeedsCompilation yes

Author Maciej Bartoszuk [aut, cre], Marek Gagolewski [aut]

Maintainer Maciej Bartoszuk <bartoszuk@rexamine.com>

Repository CRAN

Date/Publication 2018-06-21 13:59:34 UTC

R topics documented:

`SimilaR-package` *The SimilaR Package*

Description

See `SimilaR_fromDirectory()` for details.

Author(s)

Maciej Bartoszuk, Marek Gagolewski

References

SimilaR *Package homepage*, http://SimilaR.rexamine.com/

`SimilaR_fromDirectory` *SimilaR_fromDirectory*

Description

An implementation of the SimilaR algorithm - a novel R code similarity evaluation algorithm based on program dependence graphs. This version gets a path to directory, where *.R files are stored to be compared.

Usage

```
SimilaR_fromDirectory(dirname, returnType = c("data.frame", "matrix"),
  fileTypes = c("function", "file"), aggregation = c("sym", "tnorm",
  "both"))
```

Arguments

`dirname`	path to a directory with R source files to be compared.
`returnType`	`"data.frame"` or `"matrix"`; indicates the output object type (see below).
`fileTypes`	`"function"` or `"file"`; indicates which pairs of functions extracted from the source files in `dirname` should be compared. `"function"` means that every function should be compared against every other function, even if both of them are defined in the same file. `"file"` means that only the functions defined in different files should be compared.
`aggregation`	`"sym"`, `"tnorm"`, or `"both"`; specifies which model of similarity asymmetry should be used. `"sym"` means that one value of similarity is computed. `"tnorm"` means that two values are obtained: one means how much the first function is a subset of the second, and the other one means how much the second function is a subset of the first. After that they are aggregated to one value using the average. `"both"` means these two values are not aggregated.

Details

Note that depending on the "aggregation" argument, the implemented method may either return a single value, representing the overall similarity between some pair functions (like a distance between them), or two different values, one measuring how much the first function is a subset of the second, and the another one evaluating how much the second function is a subset of the first one. The user may possibly wish to aggregate these two values by some custom aggregation function.

Value

If returnType is equal to "data.frame", a data frame is returned, where every row gives the information about the similarity of a different pair of functions. Columns of the data frame are as follows:

- name1 - name of the first function in a pair. Name is constructed as follows: fileName.R functionName
- name2 - name of the second function in a pair. Name is constructed as follows: fileName.R functionName
- SimilaR - values from [0,1] interval returned by SimilaR code similarity algorithm. 1 means identical function, 0 means totally dissimilar functions. If aggregation is equal "both", two columns are returned: the one with suffix "12", which means how much the first function is a subset of the second, and the another one with suffix "21" which means how much the second function is a subset of the first one
- decision - binary value, 0 or 1. 1 means that these two functions are similar, while 0 means otherwise.

Rows in the dataframe are sorted by column SimilaR, not increasingly.

If returnType is equal to "matrix", a square matrix is returned. (i,j) position equals a similarity measure between i-th and j-th function. When aggregation is equal to "sym" or "tnorm", the matrix is symmetric. For "both" it is not symmetric and (i,j) means how much the i-th function is a subset of the second, while (j,i) means how much the j-th function is a subset of the i-th. Colnames and rownames of the matrix are names of compared functions, similarly to columns name1 and name2 in a dataframe.

References

Bartoszuk M., Ph.D. thesis, in preparation, Warsaw University of Technology, Warsaw, Poland, 2018.

Bartoszuk M., Gagolewski M., *Binary aggregation functions in software plagiarism detection*, In: *Proc. FUZZ-IEEE'17*, IEEE, 2017.

Bartoszuk M., Beliakov G., Gagolewski M., James S., *Fitting aggregation functions to data: Part II - Idempotentization*, In: Carvalho J.P. et al. (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Part II (Communications in Computer and Information Science 611)*, Springer, 2016, pp. 780-789. doi:10.1007/978-3-319-40581-0_63.

Bartoszuk M., Beliakov G., Gagolewski M., James S., *Fitting aggregation functions to data: Part I - Linearization and regularization*, In: Carvalho J.P. et al. (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Part II (Communications in Computer and Information Science 611)*, Springer, 2016, pp. 767-779. doi:10.1007/978-3-319-40581-0_62.

Bartoszuk M., Gagolewski M., *Detecting similarity of R functions via a fusion of multiple heuristic methods*, In: Alonso J.M., Bustince H., Reformat M. (Eds.), *Proc. IFSA/EUSFLAT 2015*, Atlantis Press, 2015, pp. 419-426.

Bartoszuk M., Gagolewski M., *A fuzzy R code similarity detection algorithm*, In: Laurent A. et al. (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Part III (CCIS 444)*, Springer-Verlag, Heidelberg, 2014, pp. 21-30.

Examples

```
## Typical example, we want to compare functions from different files,
## but we do not want to compare two functions from the same file.
## A result is a dataframe. There will be one value describing
## similarity level.
SimilaR_fromDirectory(system.file("testdata","data",package="SimilaR"),
                                returnType = "data.frame",
                                fileTypes="file",
                                aggregation = "sym")

## In this example we want to compare every pair of functions: even these,
## which are from the same file. A result is a dataframe. There will be
## two values describing similariy levels.
SimilaR_fromDirectory(system.file("testdata","data2",package="SimilaR"),
                      returnType = "data.frame",
                      fileTypes="function",
                      aggregation = "both")

## In this example returned value is a symmetric matrix.
SimilaR_fromDirectory(system.file("testdata","data2",package="SimilaR"),
                      returnType = "matrix",
                      fileTypes="function",
                      aggregation = "tnorm")
```

`SimilaR_fromTwoFunctions` *SimilaR_fromTwoFunctions*

Description

An implementation of the SimilaR algorithm - a novel R code similarity evaluation algorithm based on program dependence graphs. This version gets two function objects and compare them. It can be used e.g. for testing purposes.

Usage

```
SimilaR_fromTwoFunctions(function1, function2, returnType = c("data.frame",
  "matrix"), aggregation = c("sym", "tnorm", "both"))
```

Arguments

function1	a function object to compare
function2	a function object to compare
returnType	"data.frame" or "matrix"; indicates the output object type (see below).
aggregation	"sym", "tnorm", or "both"; specifies which model of similarity asymmetry should be used. "sym" means that one value of similarity is computed. "tnorm" means that two values are obtained: one means how much the first function is a subset of the second, and the other one means how much the second function is a subset of the first. After that they are aggregated to one value using the average. "both" means these two values are not aggregated.

Details

Note that depending on the "aggregation" argument, the implemented method may either return a single value, representing the overall similarity between some pair functions (like a distance between them), or two different values, one measuring how much the first function is a subset of the second, and the another one evaluating how much the second function is a subset of the first one. The user may possibly wish to aggregate these two values by some custom aggregation function.

Value

If returnType is equal to "data.frame", a data frame is returned, with just one row that gives the information about the similarity of a given pair of functions. Columns of the data frame are as follows:

- name1 - name of the first function in a pair. Name is obtained by substitute() function
- name2 - name of the second function in a pair. Name is obtained by substitute() function
- SimilaR - values from [0,1] interval returned by SimilaR code similarity algorithm. 1 means identical function, 0 means totally dissimilar functions. If aggregation is equal "both", two columns are returned: the one with suffix "12", which means how much the first function is a subset of the second, and the another one with suffix "21" which means how much the second function is a subset of the first one
- decision - binary value, 0 or 1. 1 means that these two functions are similar, while 0 means otherwise.

If returnType is equal to "matrix", a square matrix is returned. (i,j) position equals a similarity measure between i-th and j-th function. When aggregation is equal to "sym" or "tnorm", the matrix is symmetric. For "both" it is not symmetric and (i,j) means how much the i-th function is a subset of the second, while (j,i) means how much the j-th function is a subset of the i-th. Colnames and rownames of the matrix are names of compared functions, similarly to columns name1 and name2 in a dataframe. Obviously in this version of function the matrix has 2 rows and 2 columns.

References

Bartoszuk M., Ph.D. thesis, in preparation, Warsaw University of Technology, Warsaw, Poland, 2018.

Bartoszuk M., Gagolewski M., *Binary aggregation functions in software plagiarism detection*, In: *Proc. FUZZ-IEEE'17*, IEEE, 2017.

Bartoszuk M., Beliakov G., Gagolewski M., James S., *Fitting aggregation functions to data: Part II - Idempotentization*, In: Carvalho J.P. et al. (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Part II (Communications in Computer and Information Science 611)*, Springer, 2016, pp. 780-789. doi:10.1007/978-3-319-40581-0_63.

Bartoszuk M., Beliakov G., Gagolewski M., James S., *Fitting aggregation functions to data: Part I - Linearization and regularization*, In: Carvalho J.P. et al. (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Part II (Communications in Computer and Information Science 611)*, Springer, 2016, pp. 767-779. doi:10.1007/978-3-319-40581-0_62.

Bartoszuk M., Gagolewski M., *Detecting similarity of R functions via a fusion of multiple heuristic methods*, In: Alonso J.M., Bustince H., Reformat M. (Eds.), *Proc. IFSA/EUSFLAT 2015*, Atlantis Press, 2015, pp. 419-426.

Bartoszuk M., Gagolewski M., *A fuzzy R code similarity detection algorithm*, In: Laurent A. et al. (Eds.), *Information Processing and Management of Uncertainty in Knowledge-Based Systems, Part III (CCIS 444)*, Springer-Verlag, Heidelberg, 2014, pp. 21-30.

Examples

```
f1 <- function(x) {x*x}
f2 <- function(x,y) {x+y}

## A dataframe is returned: 1 row, 4 columns
SimilaR_fromTwoFunctions(f1,
                         f2,
                         returnType = "data.frame",
                         aggregation = "tnorm")

## A dataframe is returned: 1 row, 5 columns
SimilaR_fromTwoFunctions(f1,
                         f2,
                         returnType = "data.frame",
                         aggregation = "both")

## A not symmetric square matrix is returned,
## with 2 rows and 2 columns
SimilaR_fromTwoFunctions(f1,
                         f2,
                         returnType = "matrix",
                         aggregation = "both")
```

Index