Package 'LinkOrgs'

January 11, 2024

Title LinkOrgs: Algorithms for Organizational Record Linkage

Version 0.0

Authors 'Brian Libgober <libgober@gmail.com > [aut,cre], Connor Jerzak <connor.jerzak@gmail.com> [aut, cre]'

Description An R package for organizational records using the algorithms of Jerzak & Libgober (2023+). The linkage is done based on organizational names and using half a billion open collaborated records on those names from LinkedIn users. It also contains functions implementing string matching performance metrics, as well as a fast, parallized version of fuzzy string matching.

Depends R (>= 3.3.3)

License
Creative Commons Attribution-Noncommercial-No Derivative Works 4.0, for academic use only.

Encoding UTF-8

LazyData true

Maintainer 'Connor Jerzak' <connor.jerzak@gmail.com>

Imports
data.table,plyr,Rfast,stringdist,doMC,parallel,glmnet,parallel,stringr,dplyr,fastmatch,reticulate

RoxygenNote 7.2.3

R topics documented:

AssessMatchPerformance

AssessMatchPerformance

Description

Computes the true/false positive and true/false negative rates of a candidate matching based on a ground-truth (preferably human-generated) matched dataset.

Usage

```
AssessMatchPerformance(x,y,by,...)
```

Arguments

x, y	data frames to be merged
z	the merged data frame to be analyzed. Should contain by, by.x, and/or by.y as column names, depending on usage.
z_true	a reference data frame containing target/true matched dataset. Should contain by, by.x, and/or by.y as column names, depending on usage.
by, by.x, by.y	character strings specifying of the columns used for merging.

Value

ResultsMatrix A matrix containing the information on the true positive, false positive, true negative, and false negative rate, in addition to the matched dataset size. These quantities are calculated based off all possible nrow(x)*nrow(y) candidate match pairs.

Examples

```
# Create synthetic data
x_orgnames <- c("apple","oracle","enron inc.","mcdonalds corporation")
y_orgnames <- c("apple corp","oracle inc","enron","mcdonalds co")
x <- data.frame("orgnames_x"=x_orgnames)
y <- data.frame("orgnames_y"=y_orgnames)
z <- data.frame("orgnames_x"=x_orgnames[1:2], "orgnames_y"=y_orgnames[1:2])
z_true <- data.frame("orgnames_x"=x_orgnames, "orgnames_y"=y_orgnames)

# Obtain match performance data
PerformanceMatrix <- AssessMatchPerformance(x = x,
                                    y = y,
                                    z = z,
                                    z_true = z_true,
                                    by.x = "orgnames_x",
                                    by.y = "orgnames_y")
print( PerformanceMatrix )
```

`BuildML` *A primarily internal function which builds the organizational record linkage models used in Libgober and Jerzak (2023+).*

Description

A primarily internal function which builds the organizational record linkage models used in Libgober and Jerzak (2023+).

Usage

```
BuildML(...)
```

`BuildTransfer` *A primarily internal function which builds the organizational record linkage models used in Libgober and Jerzak (2023+).*

Description

A primarily internal function which builds the organizational record linkage models used in Libgober and Jerzak (2023+).

Usage

```
BuildML(...)
```

`FastFuzzyMatch` *FastFuzzyMatch*

Description

Performs fast fuzzy matching of strings based on the string distance measure specified in `DistanceMeasure`. Matching is parallelized using all available CPU cores to increase execution speed.

Usage

```
FastFuzzyMatch(x,y,by,...)
```

Arguments

`x, y`	data frames to be merged
`by, by.x, by.y`	specifications of the columns used for merging. We follow the general syntax of `base::merge`; see `?base::merge` for more details.
`...`	For additional options, see "Details".

Details

FastFuzzyMatch can automatically process the by text for each dataset. Users may specify the following options:

- Set DistanceMeasure to control algorithm for computing pairwise string distances. Options include "osa", "jaccard", "jw". See ?stringdist::stringdist for all options. (Default is "jaccard")
- Set MaxDist to control the maximum allowed distance between two matched strings
- Set AveMatchNumberPerAlias to control the maximum allowed distance between two matched strings. Takes priority over MaxDist if both specified.
- Set qgram to control the character-level q-grams used in the distance measure. (Default is 2)
- Set RemoveCommonWords to TRUE to remove common words (those appearing in > 10% of aliases). (Default is FALSE)
- Set NormalizeSpaces to TRUE to remove hanging whitespaces. (Default is TRUE)
- Set RemovePunctuation to TRUE to remove punctuation. (Default is TRUE)
- Set ToLower to TRUE to ignore case. (Default is TRUE)

Value

z The merged data frame.

Examples

```
#Create synthetic data
x_orgnames <- c("apple","oracle","enron inc.","mcdonalds corporation")
y_orgnames <- c("apple corp","oracle inc","enron","mcdonalds co")
x <- data.frame("orgnames_x"=x_orgnames)
y <- data.frame("orgnames_y"=y_orgnames)
z <- data.frame("orgnames_x"=x_orgnames[1:2], "orgnames_y"=y_orgnames[1:2])
z_true <- data.frame("orgnames_x"=x_orgnames, "orgnames_y"=y_orgnames)

# Perform merge
linkedOrgs_fuzzy <- FastFuzzyMatch(x = x,
                       y = y,
                       by.x = "orgnames_x",
                       by.y = "orgnames_y")
```

LinkOrgs *LinkOrgs*

Description

Implements the organizational record linkage algorithms of Libgober and Jerzak (2023+).

Usage

```
LinkOrgs(x, y, by ...)
```

Arguments

x, y	data frames to be merged
by, by.x, by.y	character vector(s) that specify the column names used for merging data frames x and y. The merging variables should be organizational names. See ?base::merge for more details regarding syntax.
algorithm	character; specifies which algorithm described in Libgober and Jerzak (2023+) should be used. Options are "markov", "bipartite", "ml", and "transfer". Default is "transfer", which uses a transfer-learning approach to predicting the match probability using half a billion open-collaborated recoreds along with a large fraction of the content of the Internet.
conda_env	character string; specifies a conda environment where tensorflow and related packages have been installed. Used only when algorithm='ml' or DistanceMeasure='ml'.
ReturnDiagnostics	logical; specifies whether various match-level diagnostics should be returned in the merged data frame.
...	For additional specification options, see "Details".

Details

LinkOrgs automatically processes the name text for each dataset (specified by by or by.x, and by.y. Users may specify the following options:

- Set DistanceMeasure to control algorithm for computing pairwise string distances. Options include "osa", "jaccard", "jw". See ?stringdist::stringdist for all options. Default is "jaccard". To use the combined machine learning and network methods, set algorithm to "bipartite" or "markov", and DistanceMeasure to "ml".
- Set MaxDist to control the maximum allowed distance between two matched strings
- Set MaxDist_network to control the maximum allowed distance between two matched strings in the integration with the LinkedIn network representation.
- Set AveMatchNumberPerAlias to control the maximum allowed distance between two matched strings. Takes priority over MaxDist if both specified.
- Set AveMatchNumberPerAlias_network to control the maximum allowed distance between two matched strings in the integration with the LinkedIn network representation. Takes priority over MaxDist_network if both specified.
- Set qgram to control the character-level q-grams used in the distance measure. Default is 2.
- Set RemoveCommonWords to TRUE to remove common words (those appearing in > 10% of aliases). Default is FALSE.
- Set NormalizeSpaces to TRUE to remove hanging whitespaces. Default is TRUE.
- Set RemovePunctuation to TRUE to remove punctuation. Default is TRUE.
- Set ToLower to TRUE to ignore case. Default is TRUE.

Value

z The merged data frame.

Examples

```
#Create synthetic data
x_orgnames <- c("apple","oracle","enron inc.","mcdonalds corporation")
y_orgnames <- c("apple corp","oracle inc","enron","mcdonalds co")
x <- data.frame("orgnames_x"=x_orgnames)
y <- data.frame("orgnames_y"=y_orgnames)

# Perform merge
linkedOrgs <- LinkOrgs(x = x,
                       y = y,
                       by.x = "orgnames_x",
                       by.y = "orgnames_y",
                       MaxDist = 0.6)

print( linkedOrgs )
```

`RestoreML` *RestoreML*

Description

A function, primarily for internal used, used to initialize the machine learning models used in the record linkage algorithms of Libgober and Jerzak.

Usage

```
RestoreML(...)
```

`TrainML` *TrainML*

Description

Internal function that performs the training of the machine learning models used for organizational record linkage algorithms of Libgober and Jerzak.

Usage

```
TrainML(...)
```

url2dt *url2dt*

Description

Downloads a .zip file from a URL as a data.table from a URL.

Usage

```
url2dt(url, target_extension)
```

Arguments

url	character string with the URL housing the data object.
target_extension	(default = ".csv") character string describing the target extension of the file in the downloaded .zip folder.

Details

url2dt downloads a zipped .csv file and loads it into memory based on the input URL.

Value

z The downloaded data object from the URL.

Examples

```
# Example download
my_dt <- url2dt(url="https://www.dropbox.com/s/iqf9ids77dckopf/Directory_LinkIt_bipartite_Embeddings.csv.zi
```

Index