Package 'RcmdrPlugin.temis'

September 6, 2014

Type Package

Title Graphical Integrated Text Mining Solution

Version 0.7.2

Date 2014-09-06

Imports
Rcmdr (>= 2.1-1), tcltk, tcltk2, utils, ca, R2HTML (>= 2.3.0),RColorBrewer, latticeExtra, stringi

Depends methods, tm (>= 0.6), NLP, slam, zoo, lattice

Suggests SnowballC, ROpenOf-
fice, RODBC, tm.plugin.factiva (>= 1.4),tm.plugin.lexisnexis (>= 1.1), tm.plugin.europresse (>= 1.1),tm.plugin.alceste (>=
teR

Additional_repositories http://www.omegahat.org/R

Description An R Commander plug-in providing an integrated solution to perform
a series of text mining tasks such as importing and cleaning a corpus, and
analyses like terms and documents counts, vocabulary tables, terms
co-occurrences and documents similarity measures, time series analysis,correspondence analy-
sis and hierarchical clustering. Corpora can be imported
from spreadsheet-like files, directories of raw text files, Twit-
ter queries,as well as from Dow Jones Factiva, LexisNexis, Europresse and Alceste files.

License GPL (>= 2)

URL https://r-forge.r-project.org/projects/r-temis/

BugReports https://r-forge.r-project.org/tracker/?group_id=1437

Author Milan Bouchet-Valat [aut, cre],Gilles Bastin [aut]

Maintainer Milan Bouchet-Valat <nalimilan@club.fr>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-09-06 23:13:28

R topics documented:

`caTools` *Correspondence analysis helper functions*

Description

Restrict a correspondence analysis object to some rows or columns, and get row and column contributions.

Usage

```
rowSubsetCa(obj, indices)
colSubsetCa(obj, indices)
rowCtr(obj, dim)
colCtr(obj, dim)
```

Arguments

obj	A correspondence analysis object as returned by link{ca}.
indices	An integer vector of indices of rows/columns to be kept.
dim	An integer vector of dimensions to which point contributions should be computed.

Details

These functions are used to extend the features of the ca package.

rowSubsetCa and colSubsetCa take a link{ca} object and return it, keeping only the rows/columns that were specified. These objects are only meant for direct plotting, as they do not contain the full CA results: using them for detailed analysis would be misleading.

rowCtr and colCtr return the absolute contributions of all rows/columns to the specified axes of the CA. If several dimensions are passed, the result is the sum of the contributions to each axis.

See Also

showCorpusCaDlg, plotCorpusCa, plot.ca, ca

---

corpusCaDlg *Correspondence analysis from a tm corpus*

Description

Compute a simple correspondence analysis on the document-term matrix of a tm corpus.

Details

This dialog wraps the runCorpusCa function. The function runCorpusCa runs a correspondence analysis (CA) on the document-term matrix.

If no variable is selected in the list (the default), a CA is run on the full document-term matrix (possibly skipping sparse terms, see below). If one or more variables are chosen, the CA will be based on a stacked table whose rows correspond to the levels of the variable: each cell contains the sum of occurrences of a given term in all the documents of the level. Documents that contain a NA are skipped for this variable, but taken into account for the others, if any.

In all cases, variables that have not been selected are added as supplementary rows. If at least one variable is selected, documents are also supplementary rows, while they are active otherwise.

The first slider ('sparsity') allows skipping less significant terms to use less memory, especially with large corpora. The second slider ('dimensions to retain') allows choosing the number of dimensions that will be printed, but has no effect on the computation of the correspondance analysis.

See Also

runCorpusCa, ca, meta, removeSparseTerms, DocumentTermMatrix

`corpusClustDlg` *Hierarchical clustering of a tm corpus*

Description

Hierarchical clustering of the documents of a tm corpus.

Details

This dialog allows creating a tree of the documents present in a **tm** corpus either based on its document-term matrix, or on selected dimensions of a previously run correspondence analysis (if no correspondence analysis has been performed, the relevant widgets are not available). With both methods, the dendrogram starts with all separate documents at the bottom, and progressively merges them into clusters until reaching a single group at the top.

Technically, Ward's minimum variance method is used with a Chi-squared distance: see `hclust` for details about the clustering process.

The first slider allows skipping less significant terms to use less memory with large corpora. The second allows choosing what dimensions of the correspondence analysis should be used, which helps removing noise to concentrate on identified caracteristics of the corpus.

Since the clustering by itself only returns a tree, cutting it at a given size is needed to create classes of documents: this is offered automatically after the dendrogram has been computed, and can be achieved as many times as needed thanks to the Text Mining->Hierarchical clustering->Create clusters... dialog.

See Also

hclust, dist, corpusCaDlg, removeSparseTerms, DocumentTermMatrix, createClustersDlg

---

`corpusDissimilarity` *Cross-Dissimilarity Table*

Description

Build a cross-dissimilarity table reporting Chi-squared distances from two document-term matrices of the same corpus.

Usage

```
corpusDissimilarity(x, y)
```

Arguments

x	a document-term matrix
y	a document-term matrix

Details

This function can be used to build a cross-dissimilarity table from two different variables of a corpus. It takes two versions of a document-term matrix, aggregated in different ways, and returns the Chi-squared distance between each combination of the tow matrices' rows. Thus, the resulting table has rows of x for rows, and rows of y for columns.

See Also

dissimilarityTableDlg, DocumentTermMatrix, dist

---

createClustersDlg *Cut hierarchical clustering tree into clusters*

Description

Cut a hierarchical clustering tree into clusters of documents.

Details

This dialog allows grouping the documents present in a **tm** corpus according to a previously computed hierarchical clustering tree (see corpusClustDlg). It adds a new meta-data variable to the corpus, each number corresponding to a cluster; this variable is also added to the corpusMetaData data set. If clusters were already created before, they are simply replaced.

Clusters will be created by starting from the top of the dendrogram, and going through the merge points with the highest position until the requested number of branches is reached.

A window opens to summarize created clusters, providing information about specific documents and terms for each cluster. Specific terms are those whose observed frequency in the document or level has the lowest probability under an hypergeometric distribution, based on their global frequencies in the corpus and on the number of occurrences of all terms in the considered cluster. All terms with a probability below the value chosen using the third slider are reported, ignoring terms with fewer occurrences in the whole corpus than the value of the fourth slider (these terms can often have a low probability but are too rare to be of interest). The last slider allows limiting the number of terms that will be shown for each cluster.

The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column. The definition of columns is:

"% Term/Level": the percent of the term's occurrences in all terms occurrences in the level.

"% Level/Term": the percent of the term's occurrences that appear in the level (rather than in other levels).

"Global %": the percent of the term's occurrences in all terms occurrences in the corpus.

"Level": the number of occurrences of the term in the level ("internal").

"Global": the number of occurrences of the term in the corpus.

"t value": the quantile of a normal distribution corresponding the probability "Prob.".

"Prob.": the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

Specific documents are selected using a different criterion than terms: documents with the smaller Chi-squared distance to the average vocabulary of the cluster are shown. This is a euclidean distance, but weighted by the inverse of the prevalence of each term in the whole corpus, and controlling for the documents' different lengths.

This dialog can only be used after having created a tree, which is done via the Text Mining->Hierarchical clustering->Create dendrogram... dialog.

See Also

corpusClustDlg, cutree, hclust, dendrogram

`dissimilarityTableDlg` *Documents/Variables Dissimilarity Table*

Description

Build a dissimilarity table reporting Chi-squared distances between documents and/or levels of a variable.

Details

This dialog can be used in two main ways. If "Document" or one variable is selected for both rows and columns, the one-to-one dissimilarity between all documents or levels of the variable will be reported. If a different variables are chosen for rows and for columns, a cross-dissimilarity table will be created; such a table can be used to assess whether a document or variable level is closer to another variable level.

In all cases, the reported value is the Chi-squared distance between the two documents or variable levels, computed from the total document-term matrix (aggregated for variables).

See Also

corpusDissimilarity, setCorpusVariables, meta, DocumentTermMatrix, dist

`freqTermsDlg` *List most frequent terms of a corpus*

Description

List terms with the highest number of occurrences in the document-term matrix of a corpus.

Details

This dialog allows printing the most frequent terms of the corpus. If a variable is chosen, the returned terms correspond to those with the highest total among the documents within each level of the variable. If "None (whole corpus)" is selected, the absolute frequency of the chosen terms and their percents in occurrences of all terms in the whole corpus are returned. If "Document" or a variable is chosen, details about the association of the term with documents or levels are shown:

"% Term/Level": the percent of the term's occurrences in all terms occurrences in the level.

"% Level/Term": the percent of the term's occurrences that appear in the level (rather than in other levels).

"Global %": the percent of the term's occurrences in all terms occurrences in the corpus.

"Level": the number of occurrences of the term in the level ("internal").

"Global": the number of occurrences of the term in the corpus.

"t value": the quantile of a normal distribution corresponding the probability "Prob.".

"Prob.": the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

The probability is that of observing such extreme frequencies of the considered term in the level, under an hypergeometric distribution based on its global frequency in the corpus and on the number of occurrences of all terms in the document or variable level considered. The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column.

See Also

`frequentTerms`, `setCorpusVariables`, `meta`, `restrictTermsDlg`, `termsDictionary`

`frequentTerms` *List most frequent terms of a corpus*

Description

List terms with the highest number of occurrences in the document-term matrix of a corpus, possibly grouped by the levels of a variable.

Usage

```
frequentTerms(dtm, variable = NULL, n = 25)
```

Arguments

dtm	a document-term matrix.
variable	a vector whose length is the number of rows of dtm, or NULL to report most frequent terms by document; use NA to report most frequent terms in the whole corpus.
n	the number of terms to report for each level.

Details

The probability is that of observing such extreme frequencies of the considered term in the level, under an hypergeometric distribution based on its global frequency in the corpus and on the number of occurrences of all terms in the document or variable level considered. The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column.

Value

If variable = NA, one matrix with columns "Global" and Global % (see below). Else, a list of matrices, one for each level of the variable, with seven columns:

''% Term/Level''	the percent of the term's occurrences in all terms occurrences in the level.
''% Level/Term''	the percent of the term's occurrences that appear in the level (rather than in other levels).
''Global %''	the percent of the term's occurrences in all terms occurrences in the corpus.
''Level''	the number of occurrences of the term in the level ("internal").
''Global''	the number of occurrences of the term in the corpus.
''t value''	the quantile of a normal distribution corresponding the probability "Prob.".
''Prob.''	the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

Author(s)

Milan Bouchet-Valat

See Also

specificTerms, DocumentTermMatrix

GDf-class *Class* "GDf"

Description

GUI editor for data frames

Fields

`widget`: Object of class `ANY`.

`block`: Object of class `ANY`.

`head`: Object of class `ANY`.

Methods

`get_length()`: Get the number of columns in the data frames.

`set_names(values, ...)`: Set column names.

`focus_cell(i, j)`: Give focus to a given cell.

`hide_row(i, hide)`: Hide a given row.

`hide_column(j, hide)`: Hide a given column.

`initialize(parent, items, ...)`: Initialize the widget with items.

`set_items(value, i, j, ...)`: Set the value of cells.

`get_names()`: Get column names.

`init_widget(parent)`: Initialize the widget.

`set_editable(j, value)`: Set whether a column can be edited.

`sort_bycolumn(j, decreasing)`: Set the sorting column.

`save_data(nm, where)`: Save contents to a data frame.

importCorpusDlg *Import a corpus and process it*

Description

Import a corpus, process it and extract a document-term matrix.

Details

This dialog allows creating a **tm** corpus from various sources. Once the documents have been loaded, they are processed according to the chosen settings, and a document-term matrix is extracted.

The first source, "Directory containing plain text files", creates one document for each .txt file found in the specified directory. The documents are named according to the name of the file they were loaded from. When choosing the directoty where the .txt files can be found, please note that files are not listed in the file browser, only directories, but they will be loaded nevertheless.

The second source, "Spreadsheet file", creates one document for each row of a file containg tabular data, typically an Excel (.xls) or Open Document Spreadsheet (.ods), comma-separated values (.csv) or tab-separated values (.tsv, .txt, .dat) file. One column must be specified as containing the text of the document, while the remaining columns are added as variables describing each document. For the CSV format, "," or ";" is used as separator, whichever is the most frequent in the 50 first lines of the file.

The third, fourth and fifth sources, "Factiva XML or HTML file(s)", "LexisNexis HTML file(s)" and "Europresse HTML file(s)", load articles exported from the corresponding website in the XML or HTML formats (for Factiva, the former is recommended if you can choose it). Various meta-data variables describing the articles are automatically extracted. If the corpus is split into several .xml or .html files, you can put them in the same directory and select them by holding the Ctrl key to concatenate them into a single corpus. Please note that some articles from Factiva are known to contain invalid character that trigger an error when loading. If this problem happens to you, please try to identify the problematic article, for example by removing half of the documents and retrying, until only one document is left in the corpus; then, report the problem to the Factiva Customer Service, or ask for help to the maintainers of the present package.

The sixth source, "Alceste file(s)", loads texts and variables from a single file in the Alceste format, which uses asterisks to separate texts and code variables.

The seventh source, "Twitter search", retrieves most recent tweets matching the search query and written in the specified language, up to the chosen maximum number of messages. Please note that you need to register a custom application and fill in the needed information to authenticate with the Twitter API (see `vignette("twitteR")` about OAuth authentication and https://dev.twitter.com/apps/new/ to register a new application). Due to limitations imposed by Twitter, only tweets published up to 6 or 9 days ago can be downloaded, and up to a maximum number of 1500 tweets. Search queries can notably include one or more terms that must be present together for a tweet to match the query, and/or of hashtags starting with "#"; see https://dev.twitter.com/docs/using-search if you need more complex search strings. User names, hashtags, URLs and "RT" (re-tweet) mentions are automatically removed from the corpus when computing the document-term matrix as they generally disturb the analysis. If the option to remove user names and hashtags is disabled, they will be included as standard text, i.e. "#" and "@" will be removed if the punctuation removal processing option has been enabled. The "Exclude retweets" option works by identifying tweets that contain "RT" as a separate expression; this operation can also be carried out manually later by using the "Retweet" corpus variable that is created automatically at import time.

The original texts can optionally be split into smaller chunks, which will then be considered as the real unit (called 'documents') for all analyses. In order to get meaningful chunks, texts are only splitted into paragraphs. These are defined by the import filter: when importing a directory of text files, a new paragraph starts with a line break; when importing a Factiva files, paragraphs are defined by the content provider itself, so may vary in size (heading is always a separate paragraph);

splitting has no effect when importing from a spreadsheet file. A corpus variable called "Document" is created, which identifies the original text the chunk comes from.

For all sources, a data set called `corpusVariables` is created, with one row for each document in the corpus: it contains meta-data that could be extracted from the source, if any, and can be used to enter further meta-data about the corpus. This can also be done by importing an existing data set via the Data->Load data set or Data->Import data menus. Whatever way you choose, use the Text mining->Set corpus meta-data command after that to set or update the corpus's meta-data that will be used by later analyses (see `setCorpusVariables`).

The dialog also provides a few processing options that will most likely be all run in order to get a meaningful set of terms from a text corpus. Among them, stopwords removal and stemming require you to select the language used in the corpus. If you tick "Edit stemming manually", enabled processing steps will be applied to the terms before presenting you with a list of all words originally found in the corpus, together with their stemmed forms. Terms with an empty stemmed form will be excluded from the document-term matrix; the "Stopword" column is only presented as an indication, it is not taken into account when deciding whether to keep a term.

By default, the program tries to detect the encoding used by plain text (usually .txt) and comma/tab-separated values files (.csv, .tsv, .dat...). If importation fails or the imported texts contain strange characters, specify the encoding manually (a tooltip gives suggestions based on the selected language).

Once the corpus has been imported, its document-term matrix is extracted.

References

Ingo Feinerer, Kurt Hornik, and David Meyer. Text mining infrastructure in R. Journal of Statistical Software, 25(5):1-54, March 2008. Available at http://www.jstatsoft.org/v25/i05.

Ingo Feinerer. An introduction to text mining in R. R News, 8(2):19-22, October 2008. Available at http://cran.r-project.org/doc/Rnews/Rnews_2008-2.pdf

See Also

`Corpus`, `DocumentTermMatrix`, `restrictTermsDlg`, `setCorpusVariables`, `tolower`, `removePunctuation`, `removeNumbers`, `stopwords`, `stemDocument`, `tm_map`

`inspectCorpus` *Inspect corpus*

Description

See contents of all documents in the corpus.

Details

This function opens a window with the contents of all documents in the current corpus. Note that the texts are shown as they were on import, i.e. before the processing steps (removing case, punctuation, numbers and stopwords, or stemming), which make the texts hard to read. Though, if the corpus was split, created chunks are shown separately.

See Also

importCorpusDlg

---

output *Output results to HTML file*

Description

Functions to output tables and plots resulting from analysis of the corpus to an HTML FILE.

Details

setOutputFile is automatically called the first time an attempt to save a result to the output file happens. It can also be called from the "Export results to report" menu.

openOutputFile launches the configured web browser (see browseURL) to open the current output file. It is automatically called the first time a new output file is set (i.e. when setOutputFile is run).

copyTableToOutput and copyPlotToOutput export objects to the select output HTML file, using the titles that were configured when the objects where created. For plots, a plotting device must be currently open. The graph is saved in the PNG format with a reasonably high quality. For tables, the last created table is used.

enableBlackAndWhite and disableBlackAndWhite functions can be used to produce black and white only graphics adapted for printing and publication. They affect the on-screen device as well as the plot copied to the output file, so that the plot can be checked for readability before exporting it.

HTML.list outputs a list to the HTML report, printing each element of the list right after its name. HTML.ca outputs a correspondence analysis object of class ca to the HTML report. summary.ca is a slightly modified version of summary.ca from the "ca" package to accept non-ASCII characters and not abbreviate document names and terms; it is used by HTML.ca internally.

---

plotCorpusCa *Plotting 2D maps in correspondence analysis of corpus*

Description

Graphical display of correspondence analysis of a corpus in two dimensions

Usage

```
plotCorpusCa(x, dim = c(1,2), map = "symmetric", what = c("all", "all"),
             mass = c(FALSE, FALSE), contrib = c("none", "none"),
             col = c("blue", "red"),
             col.text = c("black", "blue", "black", "red"),
             font = c(3, 4, 1, 2), pch = c(16, 1, 17, 24),
             labels = c(2, 2), arrows = c(FALSE, FALSE),
             cex = 0.75,
             xlab = paste("Dimension", dim[1]),
             ylab = paste("Dimension", dim[2]), ...)
```

Arguments

x	Simple correspondence analysis object returned by runCorpusCa
dim	Numerical vector of length 2 indicating the dimensions to plot on horizontal and vertical axes respectively; default is first dimension horizontal and second dimension vertical.
map	Character string specifying the map type. Allowed options include "symmetric" (default) "rowprincipal" "colprincipal" "symbiplot" "rowgab" "colgab" "rowgreen" "colgreen"
what	Vector of two character strings specifying the contents of the plot. First entry sets the rows and the second entry the columns. Allowed values are "all" (all available points, default) "active" (only active points are displayed) "passive" (only supplementary points are displayed) "none" (no points are displayed) The status (active or supplementary) of rows and columns is set in runCorpusCa using the options suprow and supcol.
mass	Vector of two logicals specifying if the mass should be represented by the area of the point symbols (first entry for rows, second one for columns)
contrib	Vector of two character strings specifying if contributions (relative or absolute) should be represented by different colour intensities. Available options are "none" (contributions are not indicated in the plot). "absolute" (absolute contributions are indicated by colour intensities). "relative" (relative contributions are indicated by colour intensities). If set to "absolute" or "relative", points with zero contribution are displayed in white. The higher the contribution of a point, the closer the corresponding colour to the one specified by the col option.
col	Vector of length 2 specifying the colours of row and column point symbols, by default blue for rows and red for columns. Colours can be entered in hexadecimal (e.g. "#FF0000"), rgb (e.g. rgb(1,0,0)) values or by R-name (e.g. "red").

col.text	Vector of length 4 giving the color to be used for text of labels for row active and supplementary, column active and supplementary points. Colours can be entered in hexadecimal (e.g. "#FF0000"), rgb (e.g. rgb(1,0,0)) values or by R-name (e.g. "red").
font	Vector of length 4 giving the font to be used for text labels for row active and supplementary, column active and supplementary points. See par for a list possible values.
pch	Vector of length 4 giving the type of points to be used for row active and supplementary, column active and supplementary points. See pchlist for a list of symbols.
labels	Vector of length two specifying if the plot should contain symbols only (0), labels only (1) or both symbols and labels (2). Setting labels to 2 results in the symbols being plotted at the coordinates and the labels with an offset.
arrows	Vector of two logicals specifying if the plot should contain points (FALSE, default) or arrows (TRUE). First value sets the rows and the second value sets the columns.
cex	Numeric value indicating the size of the labels text.
xlab	Title for the x axis: see title.
ylab	Title for the y axis: see title.
...	Further arguments passed to plot, to points and to text.

Details

The function plotCorpusCa makes a two-dimensional map of the object created by runCorpusCa with respect to two selected dimensions. By default the scaling option of the map is "symmetric", that is the so-called *symmetric map*. In this map both the row and column points are scaled to have inertias (weighted variances) equal to the principal inertia (eigenvalue or squared singular value) along the principal axes, that is both rows and columns are in pricipal coordinates. Other options are as follows:

- -"rowprincipal" or "colprincipal" - these are the so-called *asymmetric maps*, with either rows in principal coordinates and columns in standard coordinates, or vice versa (also known as row-metric-preserving or column-metric-preserving respectively). These maps are biplots;
- -"symbiplot" - this scales both rows and columns to have variances equal to the singular values (square roots of eigenvalues), which gives a symmetric biplot but does not preserve row or column metrics;
- -"rowgab" or "colgab" - these are asymmetric maps (see above) with rows (respectively, columns) in principal coordinates and columns (respectively, rows) in standard coordinates multiplied by the mass of the corresponding point. These are also biplots and were proposed by Gabriel & Odoroff (1990);
- -"rowgreen" or "colgreen" - these are similar to "rowgab" and "colgab" except that the points in standard coordinates are multiplied by the square root of the corresponding masses, giving reconstructions of the standardized residuals.

This function has options for sizing and shading the points. If the option `mass` is `TRUE` for a set of points, the size of the point symbol is proportional to the relative frequency (mass) of each point. If the option `contrib` is `"absolute"` or `"relative"` for a set of points, the colour intensity of the point symbol is proportional to the absolute contribution of the points to the planar display or, respectively, the quality of representation of the points in the display.

Author(s)

Oleg Nenadic (adapted from `link{plot.ca}` by Milan Bouchet-Valat)

References

Gabriel, K.R. and Odoroff, C. (1990). Biplots in biomedical research. *Statistics in Medicine*, 9, pp. 469-485.
Greenacre, M.J. (1993) *Correspondence Analysis in Practice*. Academic Press, London.
Greenacre, M.J. (1993) Biplots in correspondence Analysis, *Journal of Applied Statistics*, 20, pp. 251 - 269.

See Also

`runCorpusCa`, `corpusCaDlg`, `summary.ca`, `print.ca`, `plot3d.ca`, `pchlist`

`recodeTimeVarDlg` *Recode Date/Time Variable*

Description

Recode a date or time meta-data variable to create a new variable, for example in order to use larger time units (month, week...).

Details

This dialog allows creating a new variable from a date or time variable, by specifying a new time format in which the values of the new variable will be expressed.

Typical use cases include:

- Create a month variable from a full date: Use format "%Y-%m" to get four-digit year and two-digit month; or "%y %B" to get two-digits year and full month name.
- Create a week variable from a full date: Use format "%U" to get the week number in the year starting on Sunday, or "%W" for the week number in the year starting on Monday.
- Create a date variable from a time variable: Use format "%Y-%m-%d" to get four-digit year, two-digit month and two-digit day.

The format codes allowed are those recognized by `strptime` (see `?strptime`), in particular:

'%a' Abbreviated weekday name in the current locale. (Also matches full name.)

'%A' Full weekday name in the current locale. (Also matches abbreviated name.)

'%b' Abbreviated month name in the current locale. (Also matches full name.)

'%B' Full month name in the current locale. (Also matches abbreviated name.)

'%d' Day of the month as decimal number (01-31).

'%H' Hours as decimal number (00-23).

'%I' Hours as decimal number (01-12).

'%m' Month as decimal number (01-12).

'%M' Minute as decimal number (00-59).

'%U' Week of the year as decimal number (00-53) using Sunday as the first day 1 of the week (and typically with the first Sunday of the year as day 1 of week 1). The US convention.

'%W' Week of the year as decimal number (00-53) using Monday as the first day 1 of the week (and typically with the first Monday of the year as day 1 of week 1). The UK convention.

'%p' AM/PM indicator in the locale. Used in conjunction with '%I' and not with '%H'.

'%S' Second as decimal number (00-61).

'%y' Year without century (00-99).

'%Y' Year with century.

"Time units" are chosen automatically according to the values of the time variable: it is set to the smallest unit in which all time values can be uniquely expressed. For example, if free dates are entered, the unit will be days; if times are entered but minutes are always 0, hours will be used; finally, if times are fully specified, seconds will be used as the time unit. The chosen unit appears in the vertical axis label of the plot.

Three measures of term occurrences are provided (when no variable is selected, "category" below corresponds to the whole corpus):

- Row percent corresponds to the part of chosen term's occurrences over all terms found in a given category (i.e., the sum of word counts of all documents from the category after processing) at each time point. This conceptually corresponds to line percents, except that only the columns of the document-term matrix that match the given terms are shown.
- Column percent corresponds to the part of the chosen term's occurrences that appear in each of the documents from a given category at each time point. This measure corresponds to the strict definition of column percents.
- Absolute counts returns the relevant part of the document-term matrix, but summed for a given time point, and after grouping documents according to their category.

The rolling mean is left-aligned, meaning that the number of documents reported for a point reflects the average of the values of the points occurring *after* it. When percents of occurrences are plotted, time units with no occurrence in the corpus are not plotted, since they have no defined value (0/0, reported as `NaN`); when a rolling mean is applied, the values are simply ignored, i.e. the mean is computed over the chosen window without the missing points.

See Also

`setCorpusVariables`, `meta`, `zoo`, `xyplot`, `varTimeSeriesDlg`, `recodeTimeVarDlg`

restrictTermsDlg *Select or exclude terms*

Description

Remove terms from the document-term matrix of a corpus to exclude them from further analyses.

Details

This dialog allows to only retain specified terms when you want to concentrate your analysis on an identified vocabulary, or to exclude a few terms that are known to interfere with the analysis.

Terms that are not retained or that are excluded are removed from the document-term matrix, and are thus no longer taken into account by any operations run later, like listing terms of the corpus or computing a correspondence analysis. They are not removed from the corpus's documents.

See Also

DocumentTermMatrix, termsDictionary, freqTermsDlg

runCorpusCa *Correspondence analysis from a tm corpus*

Description

Compute a simple correspondence analysis on the document-term matrix of a tm corpus.

Usage

```
runCorpusCa(corpus, dtm = NULL, variables = NULL, sparsity = 0.9, ...)
```

Arguments

corpus	A **tm** corpus.
dtm	an optional document-term matrix to use; if missing, DocumentTermMatrix will be called on corpus to create it.
variables	a character vector giving the names of meta-data variables to aggregate the document-term matrix (see "Details" below).
sparsity	Optional sparsity threshold (between 0 and 1) below which terms should be skipped. See removeSparseTerms from tm.
...	Additional parameters passed to ca.

Details

The function runCorpusCa runs a correspondence analysis (CA) on the document-term matrix that can be extracted from a **tm** corpus by calling the DocumentTermMatrix function, or directly from the dtm object if present.

If no variable is passed via the variables argument, a CA is run on the full document-term matrix (possibly skipping sparse terms, see below). If one or more variables are chosen, the CA will be based on a stacked table whose rows correspond to the levels of the variables: each cell contains the sum of occurrences of a given term in all the documents of the level. Documents that contain a NA are skipped for this variable, but taken into account for the others, if any.

In all cases, variables that have not been selected are added as supplementary rows. If at least one variable is passed, documents are also supplementary rows, while they are active otherwise.

The sparsity argument is passed to removeSparseTerms to remove less significant terms from the document-term matrix. This is especially useful for big corpora, which matrices can grow very large, prompting ca to take up too much memory.

Value

A ca object as returned by the ca function.

See Also

ca, meta, removeSparseTerms, DocumentTermMatrix

---

setCorpusVariables *Set corpus variables*

Description

Set corpus meta-data variables from the active data set.

Details

This command creates one corpus meta-data variable from each column of the active data set. Before doing so, it erases the previously set meta-data.

The active data set may contain as many variables (columns) as needed, but must contain exactly one row for each document in the corpus, as reported at import time. For convenience, a data set containing one example variable and as many rows as required, called corpusMetaData is created after importing the corpus, and defined as the active data set. It is meant to ease entering information about the documents, but has no special meaning: the setCorpusVariables command only uses the active data set, even if it is different from this corpusMetaData stub.

All analyses performed on the corpus are based on these variables, and never on the active data set. Thus, you need to call this function every time you want to take into account changes made to the data set.

See Also

meta, importCorpusDlg

setLastTable *Save the name of last table and give a title*

Description

This function saves the name of the last created table to allow copying it to the HTML report using the "Export results to report" menu, or directly using the copyTableToOutput function.

Usage

```
setLastTable(name, title = NULL)
```

Arguments

name	The name of the table, which must correspond to an object in the global environment.
title	The title to give to the table, which will be displayed in the report, or NULL for none.

Details

The title is saved as the "title" attribute of the object called as name in the global environment. You may need to call activateMenus so that the relevant menus are enabled.

Author(s)

Milan Bouchet-Valat

See Also

copyTableToOutput

showCorpusCaDlg *Show a correspondence analysis from a tm corpus*

Description

Displays a correspondence analysis previously computed from a tm corpus.

Details

This dialog allows plotting and showing most contributive terms and documents from a previously computed correspondence analysis (see corpusCaDlg). It allows plotting any dimensions of the CA together, showing either documents, terms, or variables set on the corpus using the Text mining->Manage corpus->Set corpus variables menu.

Compared with most correpondence analyses, CAs of a corpus tend to have many points to show Thus, the dialog provides two sliders ("Number of items to plot") allowing to show only a subset of terms, documents, the most contributive to the chosen dimension. These items are the most useful to interpret the axes.

The text window shows the active items most contributive to the chosen axis, together with their position, their contribution to the inertia of the axis ("Contribution"), and the contribution of the axis to their inertia ("Quality of Representation"). (For supplementary variables or documents, depending on the parameters chosen for the CA, absolute contributions are not reported as they do not exist by definition.) The part of total inertia represented by each axis is shown first, but the rest of the window only deals with the selected axis (horizontal or vertical).

The 'Draw point symbols for' checkboxes allow representing documents, terms and variables masses (corresponding to the size of the symbols) and relative contributions (corresponding to the color intensities). See the `contrib` argument to plotCorpusCa for details.

See Also

corpusCaDlg, plotCorpusCa, runCorpusCa, ca

---

specificTerms *List terms specific of a document or level*

Description

List terms most associated (positively or negatively) with each document or each of a variable's levels.

Usage

```
specificTerms(dtm, variable, p = 0.1, n.max = 25, sparsity = 0.95, min.occ = 2)
```

Arguments

`dtm`	a document-term matrix.
`variable`	a vector whose length is the number of rows of `dtm`, or `NULL` to report specific terms by document.
`p`	the maximum probability up to which terms should be reported.
`n.max`	the maximum number of terms to report for each level.
`sparsity`	Optional sparsity threshold (between 0 and 1) below which terms should be skipped. See removeSparseTerms from tm.
`min.occ`	the minimum number of occurrences in the whole `dtm` below which terms should be skipped.

Details

Specific terms reported here are those whose observed frequency in the document or level has the lowest probability under an hypergeometric distribution, based on their global frequencies in the corpus and on the number of occurrences of all terms in the document or variable level considered. The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column.

All terms with a probability below p are reported, up to n.max terms for each category.

Value

A list of matrices, one for each level of the variable, with seven columns:

`''% Term/Level''`
: the percent of the term's occurrences in all terms occurrences in the level.

`''% Level/Term''`
: the percent of the term's occurrences that appear in the level (rather than in other levels).

`''Global %''`
: the percent of the term's occurrences in all terms occurrences in the corpus.

`''Level''`
: the number of occurrences of the term in the level ("internal").

`''Global''`
: the number of occurrences of the term in the corpus.

`''t value''`
: the quantile of a normal distribution corresponding the probability "Prob.".

`''Prob.''`
: the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

Author(s)

Milan Bouchet-Valat

See Also

frequentTerms, DocumentTermMatrix, removeSparseTerms

---

specificTermsDlg *List terms specific of a document or level*

Description

List terms most associated (positively or negatively) with each document or each of a variable's levels.

Details

Specific terms reported here are those whose observed frequency in the document or level has the lowest probability under an hypergeometric distribution, based on their global frequencies in the corpus and on the number of occurrences in the document or variable level considered. The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column.

All terms with a probability below the value chosen using the first slider are reported, ignoring terms with fewer occurrences in the whole corpus than the value of the second slider (these terms can often have a low probability but are too rare to be of interest). The last slider allows limiting the number of terms that will be shown for each level.

The result is a list of matrices, one for each level of the chosen variable, with five columns:

"% Term/Level": the percent of the term's occurrences in all terms occurrences in the level.

"% Level/Term": the percent of the term's occurrences that appear in the level (rather than in other levels).

"Global %": the percent of the term's occurrences in all terms occurrences in the corpus.

"Level": the number of occurrences of the term in the level ("internal").

"Global": the number of occurrences of the term in the corpus.

"t value": the quantile of a normal distribution corresponding the probability "Prob.".

"Prob.": the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

See Also

specificTerms, setCorpusVariables, meta, restrictTermsDlg, termsDictionary

---

subsetCorpusByTermsDlg

Subset Corpus by Terms

Description

Create a subset of the corpus by retaining only the documents which contain (or not) specified terms.

Details

This operation will restrict the corpus, document-term matrix and the "corpusVars" data set so that they only contain documents with at least the chosen number of occurrences of at least one term from the first list (occurrences are for each term separately), *and* with less than the chosen number of occurrences of each of the terms from the second list. Both conditions must be fulfilled for a document to be retained. Previously run analyses like correspondence analysis or hierarchical clustering are removed to prevent confusion.

If you choose to save the original corpus, you will be able to restore it later from the Text mining -> Subset corpus -> Restore original corpus menu. Warning: checking this option will erase an existing backup if present. Like subsetting, restoring the original corpus removes existing correspondence analysis and hierarchical clustering objects.

If you specify both terms that should and terms that should not be present, or if all documents contain a term that should be excluded, it is possible that no document matches this condition, in which case an error is produced before subsetting the corpus.

See Also

setCorpusVariables, meta, DocumentTermMatrix

---

subsetCorpusByVarDlg *Subset Corpus by Levels of a Variable*

Description

Create a subset of the corpus by retaining only the documents for which the chosen variable is equal to specified levels.

Details

This operation will restrict the corpus, document-term matrix and the "corpusVars" data set so that they only contain documents with or without specified terms. Previously run analyses like correspondence analysis or hierarchical clustering will be removed to prevent confusion.

If you choose to save the original corpus, you will be able to restore it later from the Text mining -> Subset corpus -> Restore original corpus menu. Warning: checking this option will erase an existing backup if present. Like subsetting, restoring the original corpus removes existing correspondence analysis and hierarchical clustering objects.

See Also

setCorpusVariables, meta, DocumentTermMatrix

---

termChisqDist *Show terms co-occurrences*

Description

Show terms that are the most associated with one or several reference terms.

Usage

```
termChisqDist(term, dtm, n = 5, variable = NULL)
```

Arguments

term	A character vector of length 1 corresponding to the name of a column of dtm.
dtm	A document-term matrix.
n	The number of terms to return.
variable	An optional vector of the same length as the number of rows in dtm, giving the levels by which results should be reported.

Details

This function allows printing the terms that are most associated with one or several given terms, according to the document-term matrix of the corpus. Co-occurrence is measured by the Chi-squared distance between the (column) profiles of two terms in the matrix: the smaller the distance, the more terms have similar occurrence patterns.

When a variable is selected, the operation is run separately on each sub-matrix constituted by the documents that are members of the variable level. If the term does not appear in a level, NA is returned.

See Also

termCoocDlg, DocumentTermMatrix, restrictTermsDlg, termsDictionary, freqTermsDlg

---

termCoocDlg *Show co-occurrent terms*

Description

Show terms that are the most associated with one or several reference terms.

Details

This dialog allows printing the terms that are most associated with one or several given terms, according to the document-term matrix of the corpus. Co-occurrence is measured by the Chi-squared distance between the (column) profiles of two terms in the matrix: the smaller the distance, the more terms have similar occurrence patterns.

When a variable is selected, the operation is run separately on each sub-matrix constituted by the documents that are members of the variable level. If the term does not appear in a level, NA is returned.

When several terms are entered, the operation is simply run several times separately.

See Also

termChisqDist, DocumentTermMatrix, restrictTermsDlg, termsDictionary, freqTermsDlg

termFreqDlg *Term frequencies in the corpus*

Description

Study frequencies of chosen terms in the corpus, among documents, or among levels of a variable.

Details

This dialog allows creating a table providing information about the frequency of chosen terms among documents or levels of a variable. If "None (whole corpus)" is selected, the absolute frequency of the chosen terms and their percents in occurrences of all terms in the corpus are returned. If "Document" or a variable is chosen, details about the association of the term with documents or levels are shown:

"% Term/Level": the percent of the term's occurrences in all terms occurrences in the level.

"% Level/Term": the percent of the term's occurrences that appear in the level (rather than in other levels).

"Global %": the percent of the term's occurrences in all terms occurrences in the corpus.

"Level": the number of occurrences of the term in the level ("internal").

"Global": the number of occurrences of the term in the corpus.

"t value": the quantile of a normal distribution corresponding the probability "Prob.".

"Prob.": the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

The probability is that of observing such extreme frequencies of the considered term in the level, under an hypergeometric distribution based on its global frequency in the corpus and on the number of occurrences of all terms in the document or variable level considered. The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column.

The kind of plot to be drawn is automatically chosen from the selected measure. Row percents lead to bar plots, since the total sum of shown columns (terms) doesn't add up to 100 to be drawn. Absolute counts are also represented with bar plots, so that the vertical axis reports number of occurrences.

When either several pie charts are drawn for each word, or a single word has been entered, the string "%T" in the plot title will be replaced with the name of the term. In all cases, the string "%V" will be replaced with the name of the selected variable.

See Also

termFrequencies, setCorpusVariables, meta, DocumentTermMatrix, barchart, pie

`termFrequencies` *Frequency of chosen terms in the corpus*

Description

List terms with the highest number of occurrences in the document-term matrix of a corpus, possibly grouped by the levels of a variable.

Usage

```
termFrequencies(dtm, terms, variable = NULL, n = 25, by.term = FALSE)
```

Arguments

`dtm`	a document-term matrix.
`terms`	one or more terms, i.e. column names of `dtm`.
`variable`	a vector whose length is the number of rows of `dtm`, or `NULL` to report most frequent terms by document; use `NA` to report most frequent terms in the whole corpus.
`n`	the number of terms to report for each level.
`by.term`	whether the third dimension of the array should be terms instead of levels.

Details

The probability is that of observing such extreme frequencies of the considered term in the level, under an hypergeometric distribution based on its global frequency in the corpus and on the number of occurrences of all terms in the document or variable level considered. The positive or negative character of the association is visible from the sign of the t value, or by comparing the value of the "% Term/Level" column with that of the "Global %" column.

Value

If `variable = NA`, one matrix with columns "Global" and `Global %` (see below). Else, an array with seven columns:

``% Term/Level''	the percent of the term's occurrences in all terms occurrences in the level.
``% Level/Term''	the percent of the term's occurrences that appear in the level (rather than in other levels).
``Global %''	the percent of the term's occurrences in all terms occurrences in the corpus.
``Global''	the number of occurrences of the term in the corpus.
``Level''	the number of occurrences of the term ("internal").
``t value''	the quantile of a normal distribution corresponding the probability "Prob.".
``Prob.''	the probability of observing such an extreme (high or low) number of occurrences of the term in the level, under an hypergeometric distribution.

Author(s)

Milan Bouchet-Valat

See Also

specificTerms, DocumentTermMatrix

---

termsDictionary *Dictionary of terms found in a corpus*

Description

List all of the words that were found in the corpus, and stemmed terms present in the document-term matrix, together with their number of occurrences.

Usage

```
termsDictionary(dtm, order = c("alphabetic", "occurrences"))
```

Arguments

dtm	a document-term matrix.
order	whether to sort words alphabetically, or by number of (stemmed) occurrences.

Details

Words found in the corpus before stopwords removal and stemming are printed, together with the corresponding stemmed term that was eventually added to the document-term matrix, if stemming was enabled. Occurrences found before and after stemming are also shown.

The column "Stopword?" indicates whether the corresponding word is present in the list of stopwords for the corpus language. Words that were actually removed, either automatically by stopwords removal at import time, or manually via the Text mining->Terms->Exclude terms from analysis... menu, are signalled in the "Removed?" column. All other words are present in the final document-term matrix, in their original or in their stemmed form.

See Also

DocumentTermMatrix, restrictTermsDlg, freqTermsDlg, termCoocDlg

`termTimeSeriesDlg` *Temporal Evolution of Occurrences*

Description

Variation over time of frequencies of one or several terms in the corpus, or of one term by levels of a variable.

Details

This dialog allows computing and plotting the absolute number or row/column percent of occurrences of terms over a time variable, or of one term by levels of a variable. The format used by the chosen time variable has to be specified so that it is handled correctly. The format codes allowed are those recognized by `strptime` (see `?strptime`), in particular:

'%a' Abbreviated weekday name in the current locale. (Also matches full name.)

'%A' Full weekday name in the current locale. (Also matches abbreviated name.)

'%b' Abbreviated month name in the current locale. (Also matches full name.)

'%B' Full month name in the current locale. (Also matches abbreviated name.)

'%d' Day of the month as decimal number (01-31).

'%H' Hours as decimal number (00-23).

'%I' Hours as decimal number (01-12).

'%m' Month as decimal number (01-12).

'%M' Minute as decimal number (00-59).

'%U' Week of the year as decimal number (00-53) using Sunday as the first day 1 of the week (and typically with the first Sunday of the year as day 1 of week 1). The US convention.

'%W' Week of the year as decimal number (00-53) using Monday as the first day 1 of the week (and typically with the first Monday of the year as day 1 of week 1). The UK convention.

'%p' AM/PM indicator in the locale. Used in conjunction with '%I' and not with '%H'.

'%S' Second as decimal number (00-61).

'%y' Year without century (00-99).

'%Y' Year with century.

"Time units" are chosen automatically according to the values of the time variable: it is set to the smallest unit in which all time values can be uniquely expressed. For example, if free dates are entered, the unit will be days; if times are entered but minutes are always 0, hours will be used; finally, if times are fully specified, seconds will be used as the time unit. The chosen unit appears in the vertical axis label of the plot.

Three measures of term occurrences are provided (when no variable is selected, "category" below corresponds to the whole corpus):

- Row percent corresponds to the part of chosen term's occurrences over all terms found in a given category (i.e., the sum of word counts of all documents from the category after processing) at each time point. This conceptually corresponds to line percents, except that only the columns of the document-term matrix that match the given terms are shown.
- Column percent corresponds to the part of the chosen term's occurrences that appear in each of the documents from a given category at each time point. This measure corresponds to the strict definition of column percents.
- Absolute counts returns the relevant part of the document-term matrix, but summed after grouping documents according to their category.

The rolling mean is left-aligned, meaning that the number of documents reported for a point reflects the average of the values of the points occurring *after* it. When percents of occurrences are plotted, time units with no occurrence in the corpus are not plotted, since they have no defined value (0/0, reported as `NaN`); when a rolling mean is applied, the values are simply ignored, i.e. the mean is computed over the chosen window without the missing points.

See Also

`setCorpusVariables`, `meta`, `zoo`, `xyplot`, `varTimeSeriesDlg`, `recodeTimeVarDlg`

`varCrossTableDlg` *Two-way table of corpus meta-data variables*

Description

Build a two-way contingency table from a corpus's meta-data variables, optionally plotting the result.

Details

This dialog provides a simple way of computing frequencies from a single meta-data variable of a **tm** corpus. It is merely a wrapper around different steps available from the Statistics and Plot menus, but operating on the corpus meta-data instead of the active data set.

Plots are grouped according to the variable over which percentages are built (the first one for row percent, the second one for column percent), or according to the first variable if absolute counts are plotted. Thus, one can tweak grouping by changing either the order of the variables, or the type of computed percent.

See Also

`setCorpusVariables`, `meta`, `table`, `barchart`

varTableDlg *One-way table of a corpus meta-data variable*

Description

Build a one-way contingency table from a corpus's meta-data variable, optionally plotting the result.

Details

This dialog provides a simple way of computing frequencies from a single meta-data variable of a **tm** corpus. It is merely a wrapper around different steps available from the Statistics and Plot menus, but operating on the corpus meta-data instead of the active data set.

See Also

`setCorpusVariables`, `meta`, `table`, `barchart`

varTimeSeriesDlg *Corpus Temporal Evolution*

Description

Variation of the number of documents in the corpus over time, possibly grouped by variable.

Details

This dialog allows computing and plotting the number of documents over a time variable. The format used by the chosen time variable has to be specified so that it is handled correctly. The format codes allowed are those recognized by `strptime` (see `?strptime`), in particular:

'%a' Abbreviated weekday name in the current locale. (Also matches full name.)

'%A' Full weekday name in the current locale. (Also matches abbreviated name.)

'%b' Abbreviated month name in the current locale. (Also matches full name.)

'%B' Full month name in the current locale. (Also matches abbreviated name.)

'%d' Day of the month as decimal number (01-31).

'%H' Hours as decimal number (00-23).

'%I' Hours as decimal number (01-12).

'%m' Month as decimal number (01-12).

'%M' Minute as decimal number (00-59).

'%U' Week of the year as decimal number (00-53) using Sunday as the first day 1 of the week (and typically with the first Sunday of the year as day 1 of week 1). The US convention.

'%W' Week of the year as decimal number (00-53) using Monday as the first day 1 of the week (and typically with the first Monday of the year as day 1 of week 1). The UK convention.

'%p' AM/PM indicator in the locale. Used in conjunction with '%I' and not with '%H'.

'%S' Second as decimal number (00-61).

'%y' Year without century (00-99).

'%Y' Year with century.

"Time units" are chosen automatically according to the values of the time variable: it is set to the smallest unit in which all time values can be uniquely expressed. For example, if free dates are entered, the unit will be days; if times are entered but minutes are always 0, hours will be used; finally, if times are fully specified, seconds will be used as the time unit. The chosen unit appears in the vertical axis label of the plot.

The rolling mean is left-aligned, meaning that the number of documents reported for a point reflects the average of the values of the points occurring *after* it. When percents of documents are plotted, time units with no document in the corpus are not plotted, since they have no defined value (0/0, reported as `NaN`); when a rolling mean is applied, the values are simply ignored, i.e. the mean is computed over the chosen window without the missing points.

See Also

`setCorpusVariables`, `meta`, `zoo`, `xyplot`, `varTimeSeriesDlg`, `recodeTimeVarDlg`

`vocabularyDlg` *Vocabulary Summary*

Description

Build vocabulary summary table over documents or a meta-data variable of a corpus.

Details

This dialog allows creating tables providing several vocabulary measures for each document of a corpus, or each of the categories of a corpus variable:

- total number of terms
- number and percent of unique words, i.e. of words appearing at least once
- number and percent of hapax legomena, i.e. terms appearing once and only once
- total number of words
- number and percent of long words ("long" being defined as "at least 7 characters"
- number and percent of very long words ("very long" being defined as 'at least 10 characters'
- average word length

Words are defined as the forms of two or more characters present in the texts before stemming and stopword removal. On the contrary, unique *terms* are extracted from the global document-term matrix, which means they do not include words that were removed by treatments ran at the import step, and that words different in the original text might become identical terms if stemming was

performed. This can be considered the "correct" measure, since the purpose of corpus processing is exactly that: mark different forms of the same term as similar to allow for statistical analyses.

Two different units can be selected for the analysis. If "Document" is selected, values reported for each level correspond to the mean of the values for each of its documents; a mean column for the whole corpus is also provided. If "Level" is selected, these values correspond to the sum of the number of terms for each of the categories' documents, to the percentage of terms (ratio of the summed numbers of terms) and the average word length of the level when taken as a single document. Both versions of this measure are legitimate, but prompt different interpretations that should not be confused; on the contrary, interpretation of the summed or mean number of (long) terms is immediate.

This distinction does not make sense when documents (not levels of a variable) are used as the unit of analysis: in this case, "level" in the above explanation corresponds to "document", and two columns are provided about the whole corpus. "Corpus mean" is simply the average value of measures over all documents; "Corpus total" is the sum of the number of terms, the percentage of terms (ratio of the summed numbers of terms) and the average word length in the corpus when taken as a single document. See `vocabularyTable` for more details.

See Also

`vocabularyTable`, `setCorpusVariables`, `meta`, `DocumentTermMatrix`, `table`, `barchart`

`vocabularyTable` *Vocabulary summary table*

Description

Build a table summarizing vocabulary, optionally over a variable.

Usage

```
vocabularyTable(termsDtm, wordsDtm, variable = NULL, unit = c("document", "global"))
```

Arguments

`termsDtm`	A document-term matrix containing terms (i.e. extracted from a possibly stemmed corpus).
`wordsDtm`	A document-term matrix contaning words (i.e. extracted from a plain corpus).
`variable`	A vector of the same length as `lengthDtm` giving indexes according to which categories should be defined. If `NULL`, per-document measures are returned.
`unit`	When `variable` is not `NULL`, defines the way measures are aggregated (see below).

Details

This dialog allows creating tables providing several vocabulary measures for each document or each category of documents in the corpus:

- total number of terms
- number and percent of unique terms (i.e. appearing at least once)
- number and percent of hapax legomena (i.e. terms appearing once and only once)
- total number of words
- number and percent of long words ("long" being defined as "at least seven characters"
- number and percent of very long words ("very long" being defined as "at least ten characters"
- average word length

Words are defined as the forms of two or more characters present in the texts before stemming and stopword removal. On the contrary, unique *terms* are extracted from the global document-term matrix, which means they do not include words that were removed by treatments ran at the import step, and that words different in the original text might become identical terms if stemming was performed. This can be considered the "correct" measure, since the purpose of corpus processing is exactly that: mark different forms of the same term as similar to allow for statistical analyses.

Please note that percentages for *terms* and *words* are computed with regard respectively to the total number of terms and of words, so the denominators are not the same for all measures. See `vocabularyDlg`.

When `variable` is not `NULL`, `unit` defines two different ways of aggregating per-document statistics into per-category measures:

- `document`: Values computed for each document are simply averaged for each category.
- `global`: Values are computed for each category taken as a whole: word counts are summed for each category, and ratios and average are calculated for this level only, from the summed counts.

In both cases, the "Corpus" column follows the above definition.

See Also

`vocabularyDlg`, codeDocumentTermMatrix, `table`,

Index