Package 'pubmed.mineR'

August 26, 2014

Type Package

Title Text mining of PubMed Abstracts.

Version 1.0.2

Date 2014-08-26

Author Jyoti Sharma, S.Ramachandran, Ab Rauf Shah

Maintainer S. Ramachandran <ramu@igib.in>

Description An R package for text mining of PubMed Abstracts (http://www.ncbi.nlm.nih.gov/pubmed). The algorithms are designed for two formats (text and XML) from PubMed

Depends R (>= 2.10), methods

Imports RCurl, XML, SSOAP, NCBI2R, boot, R2HTML

Suggests apcluster

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

Collate 'Abstracts-class.R' 'HGNC-class.R' 'Yearwise.R' 'Genewise.R' 'R2S4.R' 'combineabs.R' 'gene_atomization.R' 'Find_conclusion.R' 'getabs.R' 'getabsT.R' 'gethgnc.R' 'ready.R' 'readabs.R' 'removeabs.R' 'searchabsL.R' 'searchabsT.R' 'sendabs.R' 'subabs.R' 'cleanabs.R' 'word_atomizations.R' 'SentenceToken.R' 'contextSearch.R' 'uniprotfun.R' 'Pathway_Link.R' 'Pathway_Info.R' 'tdm_for_lsa.R' 'printabs.R' 'cos_sim_calc.R' 'cos_sim_calc_boot.R' 'wordscluster.R' 'whichcluster.R' 'wordsclusterview.R' 'find_intro_conc_html.R' 'cluster_words.R' 'get_original_term.R' 'input_for_find_intro_conc_html.R' 'xmlreadabs.R' 'xmlword_atomizations.R' 'xmlgene_atomizations.R'

License GPL-3

LazyLoad yes

LazyData yes

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-26 07:47:14

R topics documented:

`Abstracts-class` *Class* `"Abstracts"` *Absract Class*

Description

S4 Class with three slots Journal, Abstract, PMID to store abstracts from PubMed

Objects from the Class

Objects can be created by calls of the form `new("Abstracts", ...)`.

Slots

`Journal`: Object of class `"character"`to store Journals of the abstracts from PubMed

`Abstract`: Object of class `"character"`to store Abstracts from the PubMed

`PMID`: Object of class `"numeric"`to store PMIDs of abstracts from PubMed

Methods

No methods defined with class "Abstracts" in the signature.

Author(s)

Dr.S.Ramachandran, Ab Rauf Shah

See Also

`searchabsL` `getabs` `contextSearch` `Genewise` `Yearwise` `combineabs` `subabs` `readabs`

Examples

```
showClass("Abstracts")
```

cleanabs *To clean the result of searchabsL*

Description

It will remove the 'NONE' abstracts from the result of searchabsL.

Usage

```
cleanabs(object)
```

Arguments

object	an S4 object of class Abstracts.

Value

an S4 object of class Abstracts.

Author(s)

Jyoti Sharma

See Also

searchabsL

Examples

```
## Not run: test1 = searchabsL(abs, include=c("term1", "term2"))
test2 = cleanabs(test1)
## End(Not run)
## here 'abs' is an S4 object of class Abstracts
## 'term1', 'term2' are the searchterms
## test1 is an S4 object containing abstracts for given terms
## and test2 is an S4 object of class Abstracts containing clean abstracts of searchabsL
```

cleanabs-methods *Methods for Function* cleanabs

Description

To clean 'NONE' part of searchabsL output.

Methods

signature(object = "Abstracts") From an S4 object of class 'Abstracts' the cleanabs function is able to clean the output of searchabsL by removing the 'NONE' part of resulted abstracts.

cluster_words *To Find the highest frequency of words within clusters*

Description

Function for finding the word (term) of highest frequency within clusters.

Usage

```
cluster_words(wordscluster, n)
```

Arguments

wordscluster	an R object containing the output of wordscluster()
n	a numeric vector containing cluster numbers

Value

a list containing cluster and its highest frequency word

Author(s)

S. Ramachandran

See Also

wordscluster

Examples

```
## Not run: test = cluster_words(wordscluster, 5)
## wordscluster is an R object of wordscluster
## 5 is number of cluster
## End(Not run)
```

combineabs *To combine the abstracts*

Description

combineabs will automatically combine two abtracts of two objects.

Usage

```
combineabs(object1, object2)
```

Arguments

object1	An S4 object of class Abstracts
object2	An S4 object of class Abstracts

Details

Two objects of class 'Abstracts' are combined to return non-redundant combined abstracts. It can be used sequentially to combine many objects of class 'Abstracts'. It will also write the number of combined abstracts into a text file named "data_out.txt"

Value

An R object containing the combined abstracts, and a text file named "data_out.txt" containing the number of abstracts combined together

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: res1 = combineabs(x,y)
## here 'x', 'y' are the S4 objects of class 'Abstracts'.
```

---

combineabs-methods Abstracts *Method to Combine Abstracts*

Description

combineabs method to combine the abstracts. object1 and object2 are from Abstracts class.

Methods

signature(object1 = "Abstracts")	An S4 object of class "Abstracts"
signature(object2 = "Abstracts")	An S4 object of class "Abstracts"

common_words_new *R Data containing words which frequently in text*

Description

This dataset is used to remove common words from the abstracts. This step is used for size reduction for further data mining.

Usage

```
data(common_words_new)
```

Format

The format is: chr "common_words_new"

Details

The dataset containing common words used to remove them from the text for size reduction.

Examples

```
data(common_words_new)
```

contextSearch *For Context Search*

Description

`contextSearch` is a method to extract the sentences containing a given query term

Usage

```
contextSearch(object, y)
```

Arguments

`object`	An S4 object of Class Abstracts containing text abstracts
`y`	a character vector of term(s)

Details

It takes object of class Abstracts and query term(s) as arguments and returns a text and latex file of the sentences containing query term. The latex file can be further converted into PDF by using the system command in R i.e. system("pdflatex filename.tex"). pdflatex is a shell command in Linux to convert the latex file into PDF. In the pdf file the terms are written in bold face type to enable ease of reading

Value

contextSearch() will write two files one is a text file named "companion.txt", and other is a Latex file. If the single term is given in query then file name comes with the term name. If multiple terms are used then the file name will be "combined.tex"

Author(s)

Dr.S.Ramachandran, Jyoti Sharma

Examples

```
## Not run: contextSearch(x, "diabetes")
## here 'x' is S4 object of class 'Abstracts', and query term is 'diabetes'.
```

---

contextSearch-methods *Method for Context Search*

Description

contextSearch will search the sentence for the given term(s).

Methods

`signature(object = "Abstracts")` The object from where it will search should be an S4 object of class Abstracts

---

cos_sim_calc *To calculate the cosine similarity between terms.*

Description

cos_sim_calc calculates the cosine measure of similarity between pairs of terms from corpus.

Usage

```
cos_sim_calc(nummatrix)
```

Arguments

nummatrix	A numerical matrix for e.g. a Term Document matrix (output from tdm_for_lsa)

Details

The term document matrix is taken as input and cosine meausures of similarity between all pairs of terms are calculated.

Value

An R object and a tab delimited text file containing the similarity values between all pairs of terms.

Note

This file can be input to cytoscape directly.

Author(s)

S. Ramachandran

See Also

tdm_for_lsa

Examples

```
## Not run: x = cos_sim_calc(nummatrix)
## here nummatrix is the 'Term Document Matrix' generated from tdm_for_lsa()
```

---

cos_sim_calc_boot *Cosine Similarity Calculation by Boot Strapping*

Description

cos_sim_calc_boot allows boot strap analysis. This function should be used as argument for 'statistic' in the boot function of 'boot' package.

Usage

```
cos_sim_calc_boot(data, indices)
```

Arguments

data	Term Document Matrix generated from tdm_for_lsa function of this package. In this matrix, rows are terms and columns are abstracts.
indices	index of matrix.

Details

while calling this function we need to transpose the input tdm and can also set the number of replicates. boot package is required to call this function.

Value

It will return a matrix containing the cosine similarity of pairs of terms in the abstracts. This object is in same format as returned by the 'boot' function of 'boot' package.

Author(s)

Dr.S.Ramachandran

See Also

tdm_for_lsa

Examples

```
## Not run: test_boot = boot(data = t(nummatrix), statistic = cos_sim_calc_boot, R = 2)
## here 'nummatrix' is a Term Document Matrix, boot inbuilt function of boot package,
## R is number of replicates here it is 2 user can extend this number.
```

Find_conclusion *To find the conclusion from the abstract(s).*

Description

This function is designed for the user convinience, so that user can get the conclusion from the abstract(s) with out reading the whole abstract(s).

Usage

```
Find_conclusion(y)
```

Arguments

y	An S4 object of class 'Abstract'.

Value

A list containing conclusion of given abstract(s)

Author(s)

S.Ramachandran, Jyoti Sharma

Examples

```
## Not run: res1 = Find_conclusion(y)
##  here 'y' is an S4 object of class Abstract.
```

find_intro_conc_html *To find the introduction and conclusion from the abstracts.*

Description

it helps to fetch the introduction and conclusion part from the abstracts.

Usage

```
find_intro_conc_html(y, themes, all)
```

Arguments

y	and S4 object of class Abstracts
themes	a character vector containing terms to be search in the abstracts
all	is logical if true, will include title and author otherwise only abstracts will be considered.

Details

find_intro_conc_htmlprovide an HTML file containing space separated introduction and conclusion part from the abstracts of given query term as well as gives a link direct to PubMed for resulted PMID.

Value

an HTML file.

Author(s)

S.Ramachandran, Jyoti Sharma

See Also

input_for_find_intro_conc_html, ~~~

Examples

```
## Not run: test = find_intro_conc_html(abs, "diet")
## here 'abs' is an S4 object of class Abstracts
## and 'diet' is a term to be search from the abstracts
```

GeneToEntrez *Data containing Entrez Ids*

Description

This dataset is used in DAVID_info function of the package, and it contains the Entrez Ids for the respective genes and these Entrez Ids will be used to get information about human genes.

Usage

```
data(GeneToEntrez)
```

Format

The format is: chr "GeneToEntrez"

Examples

```
data(GeneToEntrez)
```

Genewise *To Search the number of abstracts for Genes*

Description

Genewise reports the number of abstracts for given gene(s) name(s)

Usage

```
Genewise(object, gene)
```

Arguments

object	An S4 object of class Abstracts
gene	a character vector of gene names(HGNC approved symbol)

Details

This function will report the number of abstracts containing the query gene term(s) [HGNC approved symbols], and the result is saved in a text file "dataout.txt". Genewise() will report numbers of abstracts only. The abstracts themselves for corresponding gene names can be obtained using searchabsL() and searchabsT.

Value

Genewise will return an R object containing the abstracts for given gene, and a text file named "dataout.txt" containing the number of abstracts

Author(s)

S. Ramachandran, Jyoti Sharma

Examples

```
## Not run: Genewise(x, "TLR4")
## here 'x' contains the S4 object of Abstracts.
```

Genewise-methods *method to find the abstracts for the given gene.*

Description

`Genewise` The method Genewise will automatically report the numbers of abstracts for a given gene. It will write the result in the text file named "dataout.txt"

Methods

`signature(object = "Abstracts")` This method will search in an S4 object, containiing abstracts. It will write a text file named "dataout.txt", containing the number of abstracts for the query gene terms

---

gene_atomization *To Extract Genes from the Abstracts*

Description

`gene_atomization` will automatically fetch the genes (HGNC approved Symbol) from the text and report their frequencies. presently only HGNC approved symbols are used.

Usage

```
gene_atomization(m)
```

Arguments

`m`	An S4 object of class Abstracts

Details

The function writes a text file with file name "data_table.txt". The function gene_atomization() is used to obtain the name of genes along with their frequencies of occurence.

Value

A tab delimited table containing gene name and their frequencies of occurence.

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: gene_atomization(x)
## here x is an S4 object of class 'Abstracts'containing the abstracts
```

---

getabs *To get Abstracts for a given term.*

Description

getabs will automatically fetch the abstracts containing the query term. A base function of the package pubmed.mineR.

Usage

```
getabs(object, x, y)
```

Arguments

object	An S4 object of class Abstracts
x	A character string for the term
y	logical, if TRUE, search will be case sensitive

Details

getabs() is used to find and exctract the abstracts for any given term, from the large a large corpus of abstracts. It uses regexpr based search strategy.

Value

An S4 object of class 'Abstracts', containing the result abstracts for the given term.

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: getabs(x, "term")
## x is an S4 obeject of class abstracts containing the abstracts.
```

getabs-methods getabs *To Get abstracts for a term*

Description

getabs will search for the abstracts of a given term. It is case sensitive.

Methods

signature(object = "Abstracts") This method takes three arguments, first 'object' containing data to be search, 'x', the term to be search, 'y' is logical if set "YES" will consider the case of text.

getabsT *To get Abstracts for a given term.*

Description

getabsT will automatically fetch the abstracts containing the query term.

Usage

```
getabsT(object, x, y)
```

Arguments

object	An S4 object of class Abstracts
x	A character string for the term
y	is logical, if set TRUE, search will be case sensitive.

Details

getabsT() is similar to getabs(), but it performs more specific search.

Value

An object of class 'Abstracts', containing the resulted abstracts for term.

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: getabsT(diabdata, "term")
```

getabsT-methods *To Get Abstracts*

Description

getabsT will automatically return the abstracts of a term from the data.

Methods

signature(object = "Abstracts") getabsT will search for the abstracts of a term in the data, and will automatically write the number of abstracts into a text file named "dataout.txt".

get_original_term *To get the original terms from the corpus.*

Description

get_original_term is used to get the exact term as it is present in corpus.

Usage

```
get_original_term(m, n)
```

Arguments

m	an S4 object of class Abstracts containing the corpus.
n	a list object output from the function cluster_words

Value

a list object contatining the terms.

Author(s)

S.Ramachandran

See Also

wordscluster

Examples

```
## Not run: test = get_original_term(abs, words)
## here abs is an S4 object of class Abstracts
## words is the output object of cluster_words()
```

HGNC-class *HGNC Class for package.*

Description

"HGNC"

Objects from the Class

Objects can be created by calls of the form new("HGNC", ...).

Slots

HGNCID: Object of class "character"

ApprovedSymbol: Object of class "character"

ApprovedName: Object of class "character"

Status: Object of class "character"

PreviousSymbols: Object of class "character"

Aliases: Object of class "character"

Chromosome: Object of class "character"

AccessionNumbers: Object of class "character"

RefSeqIDs: Object of class "character"

Author(s)

Dr.S.Ramachandran, Ab Rauf Shah

See Also

Abstracts

Examples

```
showClass("HGNC")
```

HGNC2UniprotID *R Data containing HGNC2UniprotID data mapping.*

Description

This dataset contains HGNC2UniprotID from Uniprot and is used in uniprotfn() function of this package, to get the information of a gene from the Uniprot.

Usage

```
data(HGNC2UniprotID)
```

Format

The format is: chr "HGNC2UniprotID"

Details

The dataset contains HGNC2UniprotID

Examples

```
data(HGNC2UniprotID)
```

HGNCdata *R Data containing HGNC data.*

Description

This dataset contains data from Human Gene Nomenclature Committe i.e HGNC ID, HGNC approved symbol, approved name, gene synonyms, chromosome no., accession numbers and RefSeq ids.

Usage

```
data(HGNCdata)
```

Format

The format is: chr "HGNCdata"

Details

The dataset contains HGNCdata

Examples

```
data(HGNCdata)
```

input_for_find_intro_conc_html *fetch the abstracts using E-utilities.*

Description

it helps in searching and fetching the abstracts from E-utilities using PMIDs.

Usage

```
input_for_find_intro_conc_html(y, all)
```

Arguments

y	an S4 object of class Abstracts
all	is logical if true, will include title and author otherwise only abstracts.

Details

it takes ans S4 object as input and uses its PMIDs to fetch the abstracts from E-utilities. The output will be used as inout for find_intro_conc_html as it contains neat data i.e. abstracts only.

Value

a list containing abstracts and PMID

Author(s)

S.Ramachandran, Jyoti Sharma

References

literature/http:/eutils.ncbi.nlm.nih.gov/

See Also

find_intro_conc_html

Examples

```
## Not run: test=input_for_find_intro_conc_html(abs)
## here 'abs' is an S4 object of class Abstracts.
```

Pathway_Info *To get the information of pathways for Genes*

Description

Pathway_Info will give the brief information about pathways in which the query gene is involved.

Usage

```
Pathway_Info(x)
```

Arguments

x	character string for HGNC approved gene symbol.

Details

Pathway_Info() fetches the information from WikiPathways through SSOAP, an Bioconductor R package. It provides the name of the pathway with the name of species, but here in this package it is human specific

Value

A list showing the information of the pathways.

Note

Pathway_Info fetches information from world wide web so it needs the package XML for parsing.

Author(s)

Jyoti Sharma

See Also

Pathway_Link

Examples

```
Pathway_Info("TLR4")
## will output the names of pathways of the given genes
```

Pathway_Link *To get the Links of the pathways for given genes*

Description

Pathway_Link will give the links of the pathways in which the query gene is involved

Usage

```
Pathway_Link(x)
```

Arguments

x	character string to search, HGNC approved gene symbol.

Value

A list with the links for the pathways. By clicking on the links we can see the pathways on WikiPathways

Note

Pathway_Link retrieves information from web so it need the package XML to parse. This function depends upon SSOAP package, and it retrieves information from Wikipathways.

Author(s)

Jyoti Sharma

See Also

Pathway_Info

Examples

```
Pathway_Link("TLR4")
## list the links of the pathways in which the gene TLR4 is involved from Wikipathways.
```

printabs *To prind the total number of abstracts in an S4 object of class Abstracts , its start and end*

Description

It gives overview of the abstracts in an S4 object of class Abstracts.

Usage

```
printabs(object)
```

Arguments

object	An S4 object of class Abstracts.

Value

prints the total number of abstracts in an S4 object with additional information.

Author(s)

S.Ramachandran

Examples

```
## Not run: printabs(res1)
## here 'res1' is an S4 object of class Abstracts.
```

R2S4 *S4 Converter*

Description

R2S4 reads tab delimited text file with headers Journal, Abstract PMID into the object of class "Abstracts".

Usage

```
R2S4(x)
```

Arguments

x	A tab delimited text file

Details

This function is necessary for the conversion of a text file into S4 object of class 'Abstracts'.

Value

An S4 object of class Abstracts

Author(s)

S.Ramachandran, Jyoti Sharma

Examples

```
## Not run: R2S4("filename.txt")
```

---

readabs *To read Abstracts*

Description

readabs will automatically read the abstracts from the pubmed file.

Usage

```
readabs(x)
```

Arguments

x	Text file of PubMed abstracts. (Abstracts downloaded from PubMed)

Details

The saved file from a general pubmed search as text file is read via readabs().

Value

An S4 object of class "Abstracts", and a text file with tab delimited headers Journal, Abstract, PMID written with file name "newabs.txt".

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: readabs("pubmed_filename.txt")
##here x is the text file of abstracts saved from PubMed.
```

ready *To Initiate the Classes.*

Description

ready will initiate the classes neccessary for other functions.

Usage

```
ready()
```

Details

This function is neccessary to initiate the classes which are needed for the implementation of other functions.

Value

classes

Author(s)

S. Ramachandran

Examples

```
## Not run: ready()
```

removeabs *To remove abstracts for the query term.*

Description

removeabswill report the number of abstracts removed for the given query term.

Usage

```
removeabs(object, x, y)
```

Arguments

object	An S4 object of class Abstracts
x	A character string for the Term
y	is logocal, if set 'TRUE' search will be case specific

Details

removeabs() finds the abstracts for the given term and remove them from the large set of abstracts.A text file of file name "dataout.txt" will be written containing the number of abstracts removed.

Value

An S4 object of class Abstracts and a text file named "dataout.txt"

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: removeabs(x, "term", TRUE)
```

---

removeabs-methods removeabs *To remove abstracts of a term from the data.*

Description

removeabs This function will search for the abstracts containing the given term to remove them from the data.

Methods

signature(object = "Abstracts") This method depicts its function, it will remove the abstracts from the data, and the number of abstracts removed will be written the text file named "dataout.txt"

---

searchabsL *To Search the abstracts of term(s) in a combination mode.*

Description

searchabsL will search for abstracts for the given term(s). Multiple combinations are allowed.

Usage

```
searchabsL(object, yr, include, restrict, exclude)
```

Arguments

object	An S4 object of class Abstracts
yr	character vector specifies the year of search
include	character vector specifies the terms contained in the abstracts.
restrict	character vector specifies the term contained in the abstracts for which search should be restricted.
exclude	character vector specifies the terms contained in the abstracts for excluding these abstracts from the search results.

Details

In the arguments except for the object all other arguments have "NONE" as default. To export or write the result of searchabsL() we use sendabs() function.

Value

An object of class Abstracts satisfying the term combinations, In addition a text file named "out.txt" reporting the number of abstracts for given query term combinations.

Author(s)

S.Ramachandran

See Also

searchabsT

Examples

```
## Not run: searchabsL(x, include="term")
searchabsL(x, yr="2013")
searchabsL(x, restrict="term")
searchabsL(x, exclude="term")
searchabsL(x, include="term", exclude="term2")
## End(Not run)
## Here x is the object of class Abstracts containing data,
## "term" is the query term to be search.
```

---

searchabsL-methods *Searching Abstracts*

Description

searchabsL will automatically search the abstracts from the data for the given terms or their combination of several terms.

Methods

`signature(object = "Abstracts")` searchabsL will search the abstracts for the given term or combinations of several terms. In this method the argument "include" uses the boolean operator 'OR' and is liberal whereas the 'restrict' and 'exclude' use the boolean operator 'AND' to specify additional filters. If the restriction to individual terms are desired then they can be individually searched and then the multiple abstracts can be combined using combineasb() function.

`searchabsT` *To Search Abstracts*

Description

`searchabsT`It is similar to searchabsL() but performs more specific search. It performs case sensitive search.

Usage

```
searchabsT(object, yr, include, restrict, exclude)
```

Arguments

`object`	An S4 object of class Abstracts
`yr`	character vector specifies the year(s) of search.
`include`	character vector specifies the term(s) for which abstracts to be searched.
`restrict`	character vector specifies the term(s) contained in the abstracts for which search should be restricted.
`exclude`	character vector specifies the term(s) contained in the abstracts for excluding these abstracts from our search results.

Details

In the arguments except the object all arguments have "NONE" as default. Use sendabs() function to write the results in a tab delimited text file.

Value

An object of class Abstracts meeting the term and the term combinations. A text file reporting the number of abstracts for the query terms and their combinations is als written with the filename "out.txt".

Author(s)

Dr.S.Ramachandran

See Also

searchabsT

Examples

```
## Not run: searchabsT(x,yr="2013")
searchabsT(x,include="term")
searchabsT(x,restrict="term")
searchabsT(x,exclude="term")
searchabsT(x,yr="2013", include="term")
## End(Not run)
## Here x is an S4 object of class Abstracts containing the abstracts to search,
## "term" is the query term to be search.
```

---

searchabsT-methods searchabsT *Searching abstracts*

Description

searchabsT will perform a specific search for the given term.

Methods

signature(object = "Abstracts") It is similar to the searchabsL method, but it is more specific than searchabsL, it is case sensitive, however searchabsL is not.

---

sendabs *To send abstracts*

Description

sendabs will send the abstracts into a tab delimited text file with the fields Journal, Abstract, and PMID.

Usage

```
sendabs(object, x)
```

Arguments

object	An S4 object of class 'Abstracts'
x	"filename.txt" to write the abstracts

Details

A general writing function for object of class 'Abstracts'

Value

A tab delimited text file with headers Journal, Abstract, PMID.

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: sendabs(x,"abs.txt")
## here 'x' is the S4 object of class 'Abstracts' and
## 'abs.txt' is the file where abstracts will be written.
```

---

sendabs-methods *To send the Data into a File*

Description

sendabs will write the data of an object of class 'Abstracts' into a tab delimited text file with header Journal, Abstract, and PMID

Methods

`signature(object = "Abstracts")` sendabs will send the data into a text file. It writes a tab delimited text file for PubMed abstracts containing Journal, Abstract, and PMID.

SentenceToken *To Tokenize the sentences*

Description

SentenceToken will tokenize abstracts into individual sentences.

Usage

```
SentenceToken(x)
```

Arguments

x	is a character string; could be an output from paste

Details

This function is necessary for extracting sentences from abstracts, used by contextSearch function. The tokenization principle follows the overall strategy as described in contextSearch

Value

A character vector of sentences

Author(s)

S.Ramachandran

Examples

```
## Not run: SentenceToken(x)
```

---

subabs *To find sub-abstracts*

Description

subabs will automatically extract the sub-abstracts from large set of abstracts.

Usage

```
subabs(object, start, end)
```

Arguments

object	An S4 object of class Abstracts
start	integer, specifies starting limit of the range to perform search
end	integer, specifies end limit of the range to perform search

Details

From a large number of asbtracts wish to extract a subset of abstracts into a separate object.

Value

An R object of class 'Abstracts' containing the extracted abstracts meeting a given range.

Author(s)

Jyoti Sharma, S.Ramachandran

Examples

```
## Not run: subabs(x,1,5)
## Here 'x' is an S4 object of class 'Abstracts',
## 1 and 5 are the start and end point respectively.
```

subabs-methods *Getting subabstracts*

Description

subabs subabs will extract the sub abstracts corresponding to a given range, from the whole data.

Methods

`signature(object = "Abstracts")` From an S4 object of class 'Abstracts' the subabs function is able to extract the abstracts corresponding to a given range.

tdm_for_lsa *create Term Document Matrix for lsa analysis*

Description

lsa package take "Term Document Matrix" as input, so it is needed to create a 'tdm' for Abstracts and `tdm_for_lsa` do the same as it find out the frequency of given term in each abstract and each abstract is considered as separate document. It prepares term document matrix of terms in the 'abstracts' corpus

Usage

```
tdm_for_lsa(object, y)
```

Arguments

`object`	An S4 object of class 'Abstracts'
`y`	character vector specifying the terms

Value

a Term Document Matrix (Numerical matrix) containing the raw frequencies of given terms in each abstract.

Author(s)

Jyoti Sharma

Examples

```
## Not run: y = c("insulin", "inflammation", "obesity")
tdm_for_lsa(diab_abs,y)
## End(Not run)
```

`uniprotfun` *To get information about gene from the UniProt.*

Description

`uniprotfun` will access the UniProt data for a given gene as per HGNC approved gene symbols

Usage

```
uniprotfun(y)
```

Arguments

`y`	HGNC approved gene symbol as character

Details

This function retrieves data from the UniProt. At present uniprotfun() works with only HGNC approved gene symbols.

Value

A text file written with filename as the 'query' name.

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: uniprotfun(x)
```

`whichcluster` *To fetch the cluster for words*

Description

`whichcluster` is used to get the cluster in which a given word (term) occurs.

Usage

```
whichcluster(clusterobject, y)
```

Arguments

`clusterobject`	an R object containing the clusters of words output by `wordscluster` function.
`y`	a character string of query term.

Value

a list containing the number of cluster under which given term occurs.

Author(s)

S.Ramachandran

See Also

wordscluster

Examples

```
## Not run: test<-whichcluster(x, "diabetes")
## here x is an R object output form wordscluster function.
## and "diabetes" is the term for which cluster number is to be searched.
## End(Not run)
```

---

wordscluster *To cluster the words*

Description

wordscluster is used to cluster the words, using the levenshtein distance concept, which are coming together in combination with either 'prefixes' or 'suffixes' or other compound words. The first word, usually of lowest length, could be 'stemmed' word in many cases drastically so, is considered as representative for that cluster.

Usage

```
wordscluster(lower, upper)
```

Arguments

lower	lower limit for characters in word. Default = 5.
upper	upper limit of characters in word. Default = 30

Details

This function is usefull for dampening the 'explotion' of words output from word_atomizations. This step enables easy examination of the terms.

Value

a list object of words clustered together and a text filenamed "resulttable.txt" with the columns cluster number, cluster size and representatives of clusters.

Note

The function may run faster when the lower limits are reduced but 'risks' producing plenty of 'runaway' situations. Their frequencies are very rare. Runaway situations. Some 'words' with part identity to other smaller words will runaway with smaller words. This event creates an unfavorable situation whereby the generated 'clusters' of words become difficult to interpret. This situation can be minimized by increasing the lower limit of word length, however at the cost of lowering computational speed. An example is: the word hypercholesterolemia runsaway with the smaller word 'lester' which could be another name.In this instance increasing the lower limit will be more usefull. Words longer than 30 characters are usually names of chemical comnpunds in IUPAC system of nomenclature.

Author(s)

S.Ramachandran, Jyoti Rani

See Also

whichcluster word_atomizations

Examples

```
## Not run:
test=wordscluster(5, 10)
## here it will start making cluster of words of length with minimum of 5 characters
## and maximum of 10 characters.

## End(Not run)
```

---

wordsclusterview *To view the words in cluster*

Description

`wordsclusterview` is used to view the words comes in cluster formed by `wordscluster` function.

Usage

```
wordsclusterview(words_cluster, all)
```

Arguments

`words_cluster`	an R object containing output of wordscluster
`all`	is logical and default is FALSE, if set TRUE including those with one member word.

Details

The first 5 words and 5 words near the median nd 5 words at the tail end are shown for clusters with more than 15 members. In case of cluster size less than 15, all the words are written in output.

Value

It returns a text file named word_cluster_view.txt

Author(s)

S.Ramachandran

See Also

wordscluster

Examples

```
## Not run: test= wordsclusterview(cluster)
# here cluster is output from wordscluster
## End(Not run)
```

---

word_atomizations *Atomization of words*

Description

word_atomizations will automatically break the whole text into words nd rank them according to their frequency of occurence.

Usage

```
word_atomizations(m)
```

Arguments

m	An S4 object of class Abstracts

Details

word_atomizations() will break down the whole text into words after removing the extra white space, punctuation marks and very common english words.

Value

A text file containing words with their frequencies

Author(s)

S. Ramachandran, Jyoti Sharma

Examples

```
## Not run: word_atomizations(x)
## here x is the object containing abstracts.
```

xmlgene_atomizations *Gene atomization of xml abstracts.*

Description

xmlgene_atomizations is used to fetch the list of genes from the xml abstracts

Usage

```
xmlgene_atomizations(m)
```

Arguments

m	an S4 object of class Abstracts, output from xmlreadabs.

Value

a list containing genes from the text with their frquency of occurence.

Author(s)

S.Ramachandran, Jyoti Sharma

See Also

xmlreadabs

Examples

```
## Not run: test = xmlgene_atomizations(xmlabs)
## xmlabs is an S4 object of class Abstracts i.e. output of xmlreadabs
```

xmlreadabs *To read the abstracts from the PubMed saved in XML format.*

Description

xmlreadabs is modified form of readabs as it reads the abstracts downloaded/saved in XML format from PubMed. This is helpful to give clean and better result after preprocessing i.e. word_atomizations, wordscluster etc.

Usage

```
xmlreadabs(file)
```

Arguments

file	an XML file saved from PubMed.

Value

an S4 object of class Abstracts containing journals, abstracts and PMID.

Author(s)

S.Ramachandran

See Also

readabs

Examples

```
## Not run: test_run = xmlreadabs("pubmed_result.xml")
## here "pubmed_result.xml" is an xml format file downloaded from PubMed.
```

---

xmlword_atomizations *Word atomizations of abstracts from xml format.*

Description

xmlword_atomizations is used to process the abstracts from PubMed in XML format.

Usage

```
xmlword_atomizations(m)
```

Arguments

m	an S4 object of class Abstracts resulted from xmlreadabs.

Value

a list containing words from the text with their frequencies.

Note

xmlword_atomizations cannot work on output of readabs.

Author(s)

S. Ramachandran

See Also

xmlreadabs

Examples

```
## Not run: test = xmlword_atomizations(xmlabs)
## here xmlabs is an S4 object i.e. output of xmlreadabs
```

Yearwise *To Search abstracts Year wise*

Description

Yearwise reports the no. of abstracts in a year.

Usage

```
Yearwise(object, year)
```

Arguments

object	An S4 object of class Abstracts.
year	a character vector specifies the year.

Details

Yearwise() is useful to find the no. of abstracts for the given year.

Value

A text file containing the no. of abstracts for given Year(s)

Author(s)

Dr.S.Ramachandran

Examples

```
## Not run: Yearwise(x, "2011") or
Yearwise(x, c("2011", "2013", "2009")
## End(Not run)
## Here 'x' is the object containing data of PubMed abstracts.
```

`Yearwise-methods` `Yearwise` *Year wise extraction of Abstracts*

Description

`Yearwise` will report the abstracts for given year(s).

Methods

`signature(object = "Abstracts")` This method "Yearwise" is written to fetch the abstracts yearly.

Index