Package 'x.ent'

May 25, 2017

Type Package

Title eXtraction of ENTity

Description Provides a tool for extracting information (entities and relations between them) in text datasets. It also emphasizes the results exploration with graphical displays. It is a rule-based system and works with hand-made dictionaries and local grammars defined by users. 'x.ent' uses parsing with Perl functions and JavaScript to define user preferences through a browser and R to display and support analysis of the results extracted. Local grammars are defined and compiled with the tool Unitex, a tool developed by University Paris Est that supports multiple languages. See ?xconfig for an introduction.

Version 1.1.7

Date 2017-05-25

Depends R (>= 3.0.0)

Imports stringr,xtable,jsonlite,ggplot2,statmod

Maintainer Tien T. Phan <phantien84@gmail.com>

License GPL-3

URL https://github.com/win-stub/x.ent

BugReports https://github.com/win-stub/x.ent/issues

SystemRequirements Perl (>= 5.0), Unitex (>= 3.0
http://www-igm.univ-mlv.fr/~unitex/)

NeedsCompilation no

Author Nicolas Turenne [aut],
Tien T. Phan [aut, cre]

Repository CRAN

Date/Publication 2017-05-24 23:59:21 UTC

R topics documented:

`add_unique` *Add a value to a current list that every value is unique*

Description

Add a value to an existing list of values. These values are unique in the list.

Usage

```
add_unique(list, value)
```

Arguments

`list`	: a list of values
`value`	: a value that we want to add to the list

Value

`list`	return a list that elements in the list aren't duplicated

Examples

```
list1= c("a","b","c")
value = "a"
list1 <- add_unique(list1,value)
```

str_count *Count words in a text*

Description

Count words of characters in the string which satisfy a regular expression

Usage

```
str_count(x, pattern, sep)
```

Arguments

x	input string
pattern	regular expression
sep	a string used to separate columns, default is "".

Value

number	return a number of words that satisfies a regular expression

Examples

```
x = "file_1:b:$:carbonate:c:dimethylsulfide:coccoliths:co2:aragonite:calcite:"
str_count(x,pattern=":co2:",sep="")
```

trim *Remove whitespace from both sides of a string*

Description

Remove all spaces from text except for single spaces between words

Usage

```
trim(x)
```

Arguments

x	is a string that we want to delete whitespace from both sides

Examples

```
str = " Hello World! "
trim(str)
```

upload_dico *Upload file*

Description

Copy file from a local folder to a folder on the system

Usage

```
upload_dico(file)
```

Arguments

file	: the path of local file

xdata *Transform the results to data frame*

Description

This is a function using transformation of results to data frame.

Usage

```
xdata(e = NULL)
xdata_value(v, sort = "a")
```

Arguments

e	a vector of a entity or a list of entities, if it is nul, it shows all entities and relations that it is configured in the configuration file
v	a entity
sort	with the function xdata_value, variable "sort" allows you to sort values following frequency or alphabetically

Details

The data frame contains the columns of the name of entity or relationship and the rows of values of named entity.

Value

data frame	return a data frame

Author(s)

Tien T. Phan

See Also

xparse call the main function of module extraction written by Perl

Examples

```
xdata() #show all entities
xdata(c("p","b")) #show two entities: "p", "b"
xdata_value("p") #show only values of entity "p"
#there are two columns "value" et "freq" in this data frame
xdata_value("p")[["value"]] #convert to a verctor
```

---

xentity *List of entities or relations*

Description

Show all entities or relations

Usage

```
xentity()
xrelation()
```

Value

list	return a list of entities or relations

See Also

xshow display results

Examples

```
xentity()
xrelation()
```

xhist *Graph* xhist

Description

The function xhist in x.ent is a graphical representation of the distribution of entities with time.

Usage

```
xhist(v = "")
```

Arguments

v	a value of entity or the relations between entities

Details

Result after calling the function xparse has the following format:

1. file_name:entity:$:list_value_found
2. ...
3. file_name:entity1:entity2:...:$$:value_e1:value_e2:....:negation

Function xhist will convert the data format above to a data frame. The histogram uses this data frame to display graphs. The data frame format:

1. column file : name of file
2. column date : (format mm.yyyy)
3. column value_date, this value is used for creating histogram
4. column visible: if visible = 1 then this record will be used in histogram

Value

This function returns a data frame so that users can check or use it to create new graphs.

dataframe	return a data frame

See Also

xplot type graphique plot
xshow display the results of extracted data
xshow display results

Examples

```
xhist() #all documents
xhist(v="colza") #only documents contain "colza"
xhist(v="colza:altise") #only documents contain a relation "colza:altise"
```

xparse *Call script Perl for extracting data from corpus*

Description

Call script Perl for extracting data from corpus. Before you run, you must configure a configuration file ini.json in the folder config as: dictionaries, graphs of grammar (Use tools Unitex for creating)...

Usage

```
xparse(json_path = "",verbose=FALSE)
```

Arguments

json_path	path of configuration file (*.json)
verbose	logical. Should R report extra information on progress? Set to TRUE by the command-line option –verbose.

Details

Input: dictionaries, grammars (build with software Unitex). Output: a result file of every entity and relation

Value

Result file includes:

comp1	data of every entity such as: file1:entity1:$:data1:data2:
comp2	data of every relation of every entity for example: file1:entity1:entity2:$$:data1:data2:1

See Also

xshow display results

Examples

```
xparse()
```

xplot *Graph* xplot

Description

Graph xplot, this graph compares the appearance of entities or relations during one period

Usage

```
xplot(v1 = "", v2 = "", t = "")
```

Arguments

v1	O or 1 entity1 value
v2	a vector of entity2 value
t	a time value, format (mm.yyyy) or inteval of time value, for example: t=c("02.2010","02.2012")

Details

Result after calling the function xparse has the following format:

1. file_name:entity:$:list_value_found
2. ...
3. file_name:entity1:entity2:...:$$:value_e1:value_e2:....:negation

Function xplot will convert the data format above to a data frame. The xplot uses this data frame to display graphs. The data frame format:

1. column file : name of file
2. column date : (format mm.yyyy)
3. column value_date, this value is used for creating graph
4. column visible: if visible = 1 then this record will be used in graph
5. column value of entite v1 or v2 or v1 combined with v2

Value

This function returns a data frame so that users can check or use it to create new graphs.

dataframe	return a data frame

See Also

xhist type graphique histogram
xprop type graphique propotion
xshow displays results of extracted data

Examples

```
xplot(v1="colza")
xplot(v1="colza",v2=c("altice","rouille"))
xplot(v1="colza",v2=c("altice","rouille"),t="09.2010")
xplot(v1="colza",v2=c("altice","rouille"),t=c("09.2010","02.2011"))
```

`xprop` *Graph* `xprop`

Description

This visualization is a type of 100% stacked histograme. The graph `xprop` shows the distribution of the relationship between entities in the corpus. The total of the bar represents 100%.

Usage

```
xprop(v1,v2,type=1)
```

Arguments

`v1`	a vector of values
`v2`	a vector of values
`type`	type of graph

Details

After calling the function `xparse`, the result has the following format:

1. file_name:entity:$:list_value_found
2. ...
3. file_name:entity1:entity2:...:$$:value_e1:value_e2:....:negation

Function `xprop` will convert the data format above to a data frame such as:

1. a list of columns that call the values of v2. Those columns will contain a value 0 or 1.
2. a column has a name "cat" - categorie.
3. a column has a name "val" - value.
   Each line discribes the relevant information between values of vector `v1` and values of vector `v2`. If there exists a relationship between a value of `v1` with a value of `v2` then the column of value v2 will be 1, the column "cat" carrying value is the value of `v2` and the column "val" has the value current of v1.

Author(s)

Tien T. Phan

See Also

xhist type graphique histogram
xplot type graphique plot

Examples

```
xprop(v1=c("chou","colza"),v2=c("mouche du chou","rouille"))
v1 = as.vector(xdata_value("p")[["value"]])
v2 = as.vector(xdata_value("b")[["value"]])
xprop(v1,v2,type=2)
```

---

xshow *Show results*

Description

Show results after calling the function xparse.

Usage

```
xshow(e=NULL,sort="a")
```

Arguments

e	an entity or a list of entities that you want display, default e = NULL => display all columns
sort	type sort of data, default sort = "a" => sorted by alphabet, sort = "f" => sorted by frequency.

Details

Show results after calling function xparse. The result file has format:

1. entity file1:entity1:$:data1:data2:data3:
2. relation file1:entity1:entity2:$$:data_e1:data_e2:negation

Author(s)

Tien T. Phan

See Also

xparse call the main function of module extraction written by Perl
xshow display results

Examples

```
xfile() #show all names of files in corpus
xshow() #all columns
xshow(e="p",sort="a") #show result of entity "p", sorted by alphabet
xshow(e="p",sort="f")
xshow(e=c("p","m"))
```

---

xtest *Test each pair relations*

Description

We recommend four testings distribution to compare two samples:

1. Kolmogorov Smirnov test
2. Wilcoxon signed rank test
3. Student's t test
4. Compare Groups of Growth Curves

Usage

```
xtest(v1, v2)
```

Arguments

v1	a vector of the first entity
v2	a vector of the second entity

Details

The function `xtest` will combine the values in the first entity with the values in the second entity, each pair relations will be looking in documents. If this relationship exists, it will bring a value 1 otherwise 0

Author(s)

Tien T. Phan

See Also

`ks.test` Kolmogorov Smirnov test
`wilcox.test` Wilcoxon signed rank test
`t.test` Student's t test
`compareGrowthCurves` Compare Groups of Growth Curves

Examples

```
#get all values of entity bioagressor
b <- as.vector(xdata_value("b")[["value"]])
xtest("colza",b)
```

Index