Package 'docxtractr'

January 9, 2019

Title Extract Data Tables and Comments from 'Microsoft' 'Word' Documents

Version 0.6.1

Maintainer Bob Rudis <bob@rud.is>

Description 'Microsoft Word' 'docx' files provide an 'XML' structure that is fairly straightforward to navigate, especially when it applies to 'Word' tables and comments. Tools are provided to determine table count/structure, comment count and also to extract/clean tables and comments from 'Microsoft Word' 'docx' documents. There is also nascent support for '.doc' files.

SystemRequirements LibreOffice (<https://www.libreoffice.org/>) required to extract data from .doc files.

URL http://gitlab.com/hrbrmstr/docxtractr

BugReports https://gitlab.com/hrbrmstr/docxtractr/issues

Encoding UTF-8

Depends R (>= 3.2.0)

License MIT + file LICENSE

LazyData true

Suggests testthat, covr

Imports tools, xml2, purrr, dplyr, utils, httr, magrittr

RoxygenNote 6.1.1

NeedsCompilation no

Author Bob Rudis [aut, cre] (<https://orcid.org/0000-0001-5670-2640>),
Mark Dulhunty [ctb],
Karlo Guidoni-Martins [ctb],
Chris Muir [aut, ctb]

Repository CRAN

Date/Publication 2019-01-09 17:00:10 UTC

R topics documented:

`assign_colnames` *Make a specific row the column names for the specified data.frame*

Description

Many tables in Word documents are in twisted formats where there may be labels or other oddities mixed in that make it difficult to work with the underlying data. This function makes it easy to identify a particular row in a scraped `data.frame` as the one containing column names and have it become the column names, removing it and (optionally) all of the rows before it (since that's usually what needs to be done).

Usage

```
assign_colnames(dat, row, remove = TRUE, remove_previous = remove)
```

Arguments

`dat`	can be any `data.frame` but is intended for use with ones retuned by this package
`row`	numeric value indicating the row number that is to become the column names
`remove`	remove row specified by `row` after making it the column names? (Default: `TRUE`)
`remove_previous`	remove any rows preceding `row`? (Default: `TRUE` but will be assigned whatever is given for `remove`).

Value

`data.frame`

See Also

docx_extract_all, docx_extract_tbl

Examples

```
# a "real" Word doc
real_world <- read_docx(system.file("examples/realworld.docx", package="docxtractr"))
docx_tbl_count(real_world)

# get all the tables
tbls <- docx_extract_all_tbls(real_world)

# make table 1 better
assign_colnames(tbls[[1]], 2)

# make table 5 better
assign_colnames(tbls[[5]], 2)
```

---

docxtractr *Extract Data Tables and Comments from 'Microsoft' 'Word' Documents*

Description

Microsoft Word 'docx" files provide an XML structure that is fairly straightforward to navigate, especially when it applies to Word tables. The 'docxtractr" package provides tools to determine table count + table structure and extract tables from Microsoft Word docx documents. It also provides tools to determine comment count and extract comments from Word 'docx" documents.

Author(s)

Bob Rudis (bob@rud.is)

---

docx_cmnt_count *Get number of comments in a Word document*

Description

Get number of comments in a Word document

Usage

```
docx_cmnt_count(docx)
```

Arguments

docx	docx object read with read_docx

Value

numeric

Examples

```
cmnts <- read_docx(system.file("examples/comments.docx", package="docxtractr"))
docx_cmnt_count(cmnts)
```

---

docx_describe_cmnts *Returns information about the comments in the Word document*

Description

Returns information about the comments in the Word document

Usage

```
docx_describe_cmnts(docx)
```

Arguments

docx	docx object read with read_docx

Examples

```
cmnts <- read_docx(system.file("examples/comments.docx", package="docxtractr"))
docx_cmnt_count(cmnts)
docx_describe_cmnts(cmnts)
```

---

docx_describe_tbls *Returns a description of all the tables in the Word document*

Description

This function will attempt to discern the structure of each of the tables in docx and print this information

Usage

```
docx_describe_tbls(docx)
```

Arguments

docx	docx object read with read_docx

Examples

```
complx <- read_docx(system.file("examples/complex.docx", package="docxtractr"))
docx_tbl_count(complx)
docx_describe_tbls(complx)
```

`docx_extract_all` *Extract all tables from a Word document*

Description

Extract all tables from a Word document

Usage

```
docx_extract_all(docx, guess_header = TRUE, preserve = FALSE,
  trim = TRUE)
```

Arguments

`docx`	docx object read with `read_docx`
`guess_header`	should the function make a guess as to the existence of a header in a table? (Default: `TRUE`)
`preserve`	preserve line breaks within a cell? Default: 'FALSE'. NOTE: This overrides 'trim'.
`trim`	trim leading/trailing whitespace (if any) in cells? (default: `TRUE`)

Value

`list` of `data.frames` or an empty `list` if no tables exist in `docx`

See Also

`assign_colnames`, `docx_extract_tbl`

Examples

```
# a "real" Word doc

real_world <- read_docx(system.file("examples/realworld.docx", package="docxtractr"))
docx_tbl_count(real_world)

# get all the tables
tbls <- docx_extract_all_tbls(real_world)
```

docx_extract_all_cmnts *Extract all comments from a Word document*

Description

Extract all comments from a Word document

Usage

```
docx_extract_all_cmnts(docx, include_text = FALSE)
```

Arguments

docx	docx object read with read_docx
include_text	if TRUE then the text associated with the comment will also be included

Value

data_frame of comment id, author & text

Examples

```
cmnts <- read_docx(system.file("examples/comments.docx", package="docxtractr"))
docx_cmnt_count(cmnts)
docx_describe_cmnts(cmnts)
docx_extract_all_cmnts(cmnts)
```

docx_extract_all_tbls *Extract all tables from a Word document*

Description

Extract all tables from a Word document

Usage

```
docx_extract_all_tbls(docx, guess_header = TRUE, preserve = FALSE,
  trim = TRUE)
```

Arguments

docx	docx object read with read_docx
guess_header	should the function make a guess as to the existence of a header in a table? (Default: TRUE)
preserve	preserve line breaks within a cell? Default: 'FALSE'. NOTE: This overrides 'trim'.
trim	trim leading/trailing whitespace (if any) in cells? (default: TRUE)

Value

list of data.frames or an empty list if no tables exist in docx

See Also

assign_colnames, docx_extract_tbl

Examples

```
# a "real" Word doc

real_world <- read_docx(system.file("examples/realworld.docx", package="docxtractr"))
docx_tbl_count(real_world)

# get all the tables
tbls <- docx_extract_all_tbls(real_world)
```

---

docx_extract_tbl *Extract a table from a Word document*

Description

Given a document read with read_docx and a table to extract (optionally indicating whether there was a header or not and if cell whitepace trimming is desired) extract the contents of the table to a data.frame.

Usage

```
docx_extract_tbl(docx, tbl_number = 1, header = TRUE,
  preserve = FALSE, trim = TRUE)
```

Arguments

docx	docx object read with read_docx
tbl_number	which table to extract (defaults to 1)
header	assume first row of table is a header row? (default; TRUE)
preserve	preserve line breaks within a cell? Default: FALSE. NOTE: This overrides trim.
trim	trim leading/trailing whitespace (if any) in cells? (default: TRUE)

Value

data.frame

See Also

docx_extract_all, docx_extract_tbl, assign_colnames

Examples

```
doc3 <- read_docx(system.file("examples/data3.docx", package="docxtractr"))
docx_extract_tbl(doc3, 3)

intracell_whitespace <- read_docx(system.file("examples/preserve.docx", package="docxtractr"))
docx_extract_tbl(intracell_whitespace, 2, preserve=FALSE)
docx_extract_tbl(intracell_whitespace, 2, preserve=TRUE)
```

---

docx_tbl_count *Get number of tables in a Word document*

Description

Get number of tables in a Word document

Usage

```
docx_tbl_count(docx)
```

Arguments

docx	docx object read with read_docx

Value

numeric

Examples

```
complx <- read_docx(system.file("examples/complex.docx", package="docxtractr"))
docx_tbl_count(complx)
```

---

mcga *Make Column Names Great Again*

Description

Remove punctuation and spaces and turn them to underscores plus convert to lower case.

Usage

```
mcga(tbl)
```

Arguments

tbl	a data.frame-like object

Value

whatver class x was but with truly great, really great column names. They're amazing. Trust me. They'll be incredible column names once we're done.

Examples

```
real_world <- read_docx(system.file("examples/realworld.docx", package="docxtractr"))
tbls <- docx_extract_all_tbls(real_world)
mcga(assign_colnames(tbls[[1]], 2))
```

---

print.docx *Display information about the document*

Description

Display information about the document

Usage

```
## S3 method for class 'docx'
print(x, ...)
```

Arguments

x	docx object
...	ignored

---

read_docx *Read in a Word document for table extraction*

Description

Local file path or URL pointing to a .docx file. Can also take .doc file as input if LibreOffice is installed (see https://www.libreoffice.org/ for more info and to download).

Usage

```
read_docx(path, track_changes = NULL)
```

Arguments

path	path to the Word document
track_changes	if not NULL (the default) then must be one of "accept" or "reject" which will, respectively, accept all or reject all changes. NOTE: this functionality relies on the pandoc utility being available on the system PATH. Both system PATH and the RSTUDIO_PANDOC (RStudio ships with a copy of pandoc) environment variables will be checked. If no pandoc binary is found then a warning will be issued and the document will be read without integrating or ignoring any tracked changes. The original Word document *will not be modified* and this feature *only works* with docx files.

Examples

```
doc <- read_docx(system.file("examples/data.docx", package="docxtractr"))
class(doc)

doc <- read_docx(
  system.file("examples/trackchanges.docx", package="docxtractr"),
  track_changes = "accept"
)

## Not run:
# from a URL

## End(Not run)
```

---

set_libreoffice_path *Point to Local soffice.exe File*

Description

Function to set an option that points to the local LibreOffice file soffice.exe.

Usage

```
set_libreoffice_path(path)
```

Arguments

path	path to the LibreOffice soffice file

Details

For a list of possible file path locations for soffice.exe, see https://github.com/hrbrmstr/docxtractr/issues/5#issuecomment-233181976

Value

Returns nothing, function sets the option variable `path_to_libreoffice`.

Examples

```
## Not run:
set_libreoffice_path("local/path/to/soffice.exe")

## End(Not run)
```

Index