Package 'tm.plugin.mail'

August 16, 2018

Title Text Mining E-Mail Plug-in

Version 0.2-1

Date 2018-08-16

Imports NLP (>= 0.1-2), tm (>= 0.6-1)

Description A plug-in for the tm text mining framework providing mail handling functionality.

License GPL-3

NeedsCompilation no

Author Ingo Feinerer [aut] (<https://orcid.org/0000-0001-7656-8338>),
Wolfgang Mauerer [aut],
Kurt Hornik [aut, cre] (<https://orcid.org/0000-0003-4198-9911>)

Maintainer Kurt Hornik <Kurt.Hornik@R-project.org>

Repository CRAN

Date/Publication 2018-08-16 09:32:44 UTC

R topics documented:

convert_mbox_eml *Convert E-Mails From mbox Format To eml Format*

Description

Convert e-mails from mbox (i.e., several mails in a single box) format to eml (i.e., every mail in a single file) format.

Usage

```
convert_mbox_eml(mbox, dir)
```

Arguments

mbox	A character or connection describing the mbox location.
dir	A character describing the output directory.

Value

No explicit return value. As a side product the directory `dir` contains the e-mails in eml format.

Author(s)

Ingo Feinerer

MailDocument *E-Mail Documents*

Description

Create electronic mail documents.

Usage

```
MailDocument(x,
             author = character(),
             datetimestamp = as.POSIXlt(Sys.time(), tz = "GMT"),
             description = character(),
             header = character(),
             heading = character(),
             id = character(),
             language = character(),
             origin = character(),
             ...,
             meta = NULL)
```

Arguments

x	A character giving the text content.
author	a character or an object of class person giving the author names.
datetimestamp	an object of class POSIXt or a character string giving the creation date/time information. If a character string, exactly one of the ISO 8601 formats defined by http://www.w3.org/TR/NOTE-datetime should be used. See parse_ISO_8601_datetime in package **NLP** for processing such date/time information.
description	a character giving a description.
header	a character giving the mail header.
heading	a character giving the title or a short heading.
id	a character giving a unique identifier.
language	a character giving the language (preferably as IETF language tags, see language in package **NLP**).
origin	a character giving information on the source and origin.
...	user-defined document metadata tag-value pairs.
meta	a named list or NULL (default) giving all metadata. If set, all other metadata arguments are ignored.

Value

An object inheriting from MailDocument, PlainTextDocument, and TextDocument.

Author(s)

Ingo Feinerer

MBoxSource *Mailbox Source*

Description

Create a mailbox source.

Usage

```
MBoxSource(mbox, encoding = "")
```

Arguments

mbox	A character string giving the path or URL to a mailbox stored in "mbox" format.
encoding	A character string describing the current encoding. Passed to iconv to convert the input to "UTF-8".

Details

A *mailbox source* interprets each e-mail stored in the mailbox as a document.

Value

An object inheriting from `MBoxSource`, `SimpleSource`, and `Source`.

See Also

`Encoding` and `iconv` on encodings.

`readMail` *Read In an E-Mail Document*

Description

Return a function which reads in an electronic mail document.

Usage

```
readMail(DateFormat = character())
```

Arguments

`DateFormat`	A character vector giving date-time formats for the "Date" header field in the mail document. By default, the "basic" formats of RFC 5322 are tried.

Details

Formally this function is a function generator, i.e., it returns a function (which reads in a mail document) with a well-defined signature, but can access passed over arguments (e.g., the "Date" header format) via lexical scoping.

Value

A `function` with the following formals:

`elem` a named list with the component `content` which must hold the document to be read in.

`language` a string giving the language.

`id` a character giving a unique identifier for the created text document.

The function returns a `MailDocument` representing the text and metadata extracted from `elem$content`. The argument `id` is used as fallback if no corresponding metadata entry is found in `elem$content`.

Author(s)

Ingo Feinerer

See Also

Reader for basic information on the reader infrastructure employed by package **tm**.

strptime for date-time format specifications.

RFC 5322 (https://tools.ietf.org/html/rfc5322).

Examples

```
require("tm")
newsgroup <- system.file("mails", package = "tm.plugin.mail")
news <- VCorpus(DirSource(newsgroup),
                readerControl = list(reader = readMail))
inspect(news)
```

---

removeCitation *Remove E-Mail Citations*

Description

Remove citations, i.e., lines beginning with >, from an e-mail message.

Usage

```
## S3 method for class 'MailDocument'
removeCitation(x, ...)
```

Arguments

x	A mail document.
...	the argument removeQuoteHeader (default FALSE) giving a logical indicating if the quotation header (of the type "On *date*, *author* wrote:") that proceeds the quoted message should be removed.

Author(s)

Ingo Feinerer

See Also

removeMultipart to remove non-text parts from multipart e-mail messages, and removeSignature to remove signature lines from e-mail messages.

Examples

```
require("tm")
newsgroup <- system.file("mails", package = "tm.plugin.mail")
news <- VCorpus(DirSource(newsgroup),
                readerControl = list(reader = readMail))
news[[8]]
removeCitation(news[[8]])
removeCitation(news[[8]], removeQuoteHeader = TRUE)
```

`removeMultipart` *Remove Non-Text Parts From E-Mails*

Description

Remove non-text parts from multipart e-mail messages.

Usage

```
## S3 method for class 'MailDocument'
removeMultipart(x, ...)
```

Arguments

`x`	A mail document.
`...`	Not used.

Author(s)

Ingo Feinerer

See Also

`removeCitation` to remove e-mail citations, and `removeSignature` to remove signature lines from e-mail messages.

`removeSignature` *Remove E-Mail Signatures*

Description

Remove signature lines from an e-mail message.

Usage

```
## S3 method for class 'MailDocument'
removeSignature(x, ...)
```

Arguments

x	A mail document.
...	the argument marks giving a character of signature identifications marks (in form of regular expression patterns). Note that the official signature start mark -- (dash dash blank) is always considered.

Author(s)

Ingo Feinerer

See Also

removeCitation to remove e-mail citations, and removeMultipart to remove non-text parts from multipart e-mail messages.

Examples

```
require("tm")
newsgroup <- system.file("mails", package = "tm.plugin.mail")
news <- VCorpus(DirSource(newsgroup),
                readerControl = list(reader = readMail))
news[[7]]
removeSignature(news[[7]], marks = "^[+]-*[+]$")
```

---

threads *E-Mail Threads*

Description

Extract threads (i.e., chains of messages on a single subject) from e-mail documents.

Usage

```
threads(x)
```

Arguments

x	A corpus consisting of e-mails (MailDocuments).

Details

This function uses a one-pass algorithm for extracting the thread information by inspecting the "References" header. Some mails (e.g., reply mails appearing before their corresponding base mails) might not be tagged correctly.

Value

A list with the two named components ThreadID and ThreadDepth, listing a thread and the level of replies for each mail in the corpus x.

Examples

```
require("tm")
newsgroup <- system.file("mails", package = "tm.plugin.mail")
news <- VCorpus(DirSource(newsgroup),
                readerControl = list(reader = readMail))
vapply(news, meta, "id", FUN.VALUE = "")
lapply(news, function(x) meta(x, "header")$References)
(info <- threads(news))
lengths(split(news, info$ThreadID))
```

Index