

# Package ‘tm.plugin.mail’

July 2, 2014

**Title** Text Mining E-Mail Plug-In

**Version** 0.1

**Date** 2014-04-24

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

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

**License** GPL-3

**Author** Ingo Feinerer [aut, cre], Wolfgang Mauerer [aut]

**Maintainer** Ingo Feinerer <feinerer@logic.at>

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-06-11 15:06:17

## R topics documented:

convert_mbox_eml . . . . .	2
MailDocument . . . . .	2
MBoxSource . . . . .	3
readMail . . . . .	4
removeCitation . . . . .	5
removeMultipart . . . . .	6
removeSignature . . . . .	6
threads . . . . .	7
<b>Index</b>	<b>8</b>

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(0),  
  datetimestamp = as.POSIXlt(Sys.time(), tz = "GMT"),  
  description = character(0),  
  header = character(0),  
  heading = character(0),  
  id = character(0),  
  language = character(0),  
  origin = character(0),  
  ...,  
  meta = NULL)
```

**Arguments**

x	A character giving the text content.
author	a character or an object of class <a href="#">person</a> giving the author names.
datetimestamp	an object of class <a href="#">POSIXlt</a> giving the creation date and time.
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).
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. It is passed to <a href="#">iconv</a> 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 = "%d %B %Y %H:%M:%S")
```

**Arguments**

`DateFormat` A character giving the format of the “Date” header in the mail document.

**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 signature `elem, language, id`:

`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 format specifications.

## Examples

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

---

removeCitation	<i>Remove E-Mail Citations</i>
----------------	--------------------------------

---

## 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 <i>date</i> , <i>author</i> wrote:”) that precedes 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))
sapply(news, function(x) meta(x, "id"))
lapply(news, function(x) grep("References", attr(x, "Header"), value = TRUE))
threads(news)
```

# Index

convert\_mbox\_eml, 2

Encoding, 4

iconv, 3, 4

MailDocument, 2, 4

MBoxSource, 3

person, 3

PlainTextDocument, 3

POSIXlt, 3

Reader, 4

readMail, 4

removeCitation, 5, 6, 7

removeMultipart, 5, 6, 7

removeSignature, 5, 6, 6

SimpleSource, 4

Source, 4

strptime, 4

TextDocument, 3

threads, 7