Package 'tm.plugin.factiva'

May 8, 2026

Type Package

Title Import Articles from 'Factiva' Using the 'tm' Text Mining Framework

Version 1.8.1

Date 2025-03-28

Imports NLP, tm (>= 0.7-2), xml2, rvest

Description Provides a 'tm' Source to create corpora from articles exported from the Dow Jones 'Factiva' content provider as XML or HTML files. It is able to read both text content and meta-data information (including source, date, title, author, subject, geographical coverage, company, industry, and various provider-specific fields).

License GPL (>= 2)

URL https://github.com/nalimilan/R.TeMiS

BugReports https://github.com/nalimilan/R.TeMiS/issues

NeedsCompilation no

Author Milan Bouchet-Valat [aut, cre],
Grigorij Ljubownikow [ctb],
Juliane Krueger [ctb],
Tom Nicholls [ctb]

Maintainer Milan Bouchet-Valat <nalimilan@club.fr>

Repository CRAN

Date/Publication 2025-03-28 23:10:05 UTC

Contents

`tm.plugin.factiva-package` *A plug-in for the tm text mining framework to import articles from Factiva*

Description

This package provides a tm Source to create corpora from articles exported from Dow Jones's Factiva content provider as XML or HTML files.

Details

Typical usage is to create a corpus from a XML or HTML files exported from Factiva (here called `myFactivaArticles.xml`). Setting `language=NA` allows the language to be set automatically from the information provided by Factiva:

```
# Import corpus
source <- FactivaSource("myFactivaArticles.xml")
corpus <- Corpus(source, list(language=NA))

# See how many articles were imported
corpus

# See the contents of the first article and its meta-data
inspect(corpus[1])
meta(corpus[[1]])
```

Currently, only HTML files saved in French are supported. Please send the maintainer examples of Factiva files in your language if you want it to be supported.

See `FactivaSource` for more details and real examples.

Author(s)

Milan Bouchet-Valat <nalimilan@club.fr>

References

https://global.factiva.com/

FactivaSource *Factiva Source*

Description

Construct a source for an input containing a set of articles exported from Factiva in the XML or HTML formats.

Usage

```
FactivaSource(x, encoding = "UTF-8",
              format = c("auto", "XML", "HTML"))
```

Arguments

x	Either a character identifying the file or a connection.
encoding	A character giving the encoding of x, only used for HTML files. It will be ignored unless the HTML input does not include this information, which should normally not happen with files exported from Factiva.
format	The format of the file or connection identified by x (see "Details").

Details

This function can be used to import both XML and HTML files. If format is set to "auto" (the default), the file extension is used to guess the format: if the file name ends with ".xml" or ".XML", XML is assumed; else, the file is assumed to be in the HTML format.

It is advised to export articles from Factiva in the XML format rather than in HTML when possible, since the latter does not provide completely clean information. In particular, dates are not guaranteed to be parsed correctly if the machine from which the HTML file was exported uses a locale different from that of the machine where it is read.

The following screencast illustrates how to export articles in the correct HTML format from the Factiva website: https://rtemis.hypotheses.org/files/2017/02/Factiva-animated-tutorial.gif. **Do note that by not following this procedure, you will obtain a HTML file which cannot be imported by this package.**

This function imports the body of the articles, but also sets several meta-data variables on individual documents:

- datetimestamp: The publication date.
- heading: The title of the article.
- origin: The newspaper the article comes from.
- edition: The (local) variant of the newspaper.
- section: The part of the newspaper containing the article.
- subject: One or several keywords defining the subject.
- company: One or several keywords identifying the covered companies.

- `industry`: One or several keywords identifying the covered industries.
- `infocode`: One or several Information Provider Codes (IPC).
- `infodesc`: One or several Information Provider Descriptions (IPD).
- `coverage`: One or several keywords identifying the covered regions.
- `page`: The number of the page on which the article appears (if applicable).
- `wordcount`: The number of words in the article.
- `publisher`: The publisher of the newspaper.
- `rights`: The copyright information associated with the article.
- `language`: This information is set automatically if `readerControl = list(language = NA)` is passed (see the example below). Else, the language specified manually is set for all articles. If omitted, the default, "en", is used.

Value

An object of class `XMLSource` which extends the class `Source` representing set of articles from Factiva.

Note

It has been found that some Factiva articles contain unescaped characters that are not authorized in XML files. If such articles are included in the input you are trying to import, the XML parser will fail printing a few error messages, and the corpus will not be created at all.

If you experience this bug, please report this to the Factiva Customer Service, which will fix the incriminated article; feel free to ask the maintainer of the present package if needed. In the meantime, you can exclude the problematic article from the XML file: to identify it, proceed by exporting only one half of the original corpus at a time, as many times as needed, and see when it fails; you will eventually find the culprit. (If you know XML, you can use an XML validator to find the relevant part of the file, and fix it by hand.)

Author(s)

Milan Bouchet-Valat

See Also

`readFactivaXML` and `readFactivaHTML` for the functions actually parsing individual articles.

`getSources` to list available sources.

Examples

```
## Not run:
    ## For an XML file
    library(tm)
    file <- system.file("texts", "reut21578-factiva.xml",
                        package = "tm.plugin.factiva")
    source <- FactivaSource(file)
    corpus <- Corpus(source, readerControl = list(language = NA))
```

```
    # See the contents of the documents
    inspect(corpus)

    # See meta-data associated with first article
    meta(corpus[[1]])

## End(Not run)

    ## For an HTML file
    library(tm)
    file <- system.file("texts", "factiva_test.html",
                        package = "tm.plugin.factiva")
    source <- FactivaSource(file)
    corpus <- Corpus(source, readerControl = list(language = NA))

    # See the contents of the documents
    inspect(corpus)

    # See meta-data associated with first article
    meta(corpus[[1]])
```

`readFactiva` *Read in a Factiva article in XML or HTML formats*

Description

Read in an article exported from Factiva in XML or HTML formats.

Usage

```
readFactivaXML(elem, language, id)
readFactivaHTML(elem, language, id)
```

Arguments

`elem`	A `list` with the named element `content` which must hold the document to be read in.
`language`	A `character` vector giving the text's language. If set to `NA`, the language will automatically be set to the value reported in the document (which is usually correct).
`id`	A `character` vector representing a unique identification string for the returned text document.

Value

A `PlainTextDocument` with the contents of the article and the available meta-data set.

Author(s)

Milan Bouchet-Valat

See Also

`getReaders` to list available reader functions.

Index