

Package ‘boilerpipeR’

January 27, 2015

Version 1.2.2

Date 2014-08-20

Title Interface to the boilerpipe Java library by Christian Kohlschutter (<http://code.google.com/p/boilerpipe/>)

Author See AUTHORS file.

Maintainer Mario Annau <mario.annau@gmail.com>

Imports rJava

Suggests RCurl

Description Generic Extraction of main text content from HTML files; removal of ads, sidebars and headers using the boilerpipe Java library. The extraction heuristics from boilerpipe show a robust performance for a wide range of web site templates.

License Apache License (== 2.0)

URL <https://github.com/mannau/boilerpipeR>

BugReports <https://github.com/mannau/boilerpipeR/issues>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-08-21 06:42:22

R topics documented:

boilerpipeR-package	2
ArticleExtractor	2
ArticleSentencesExtractor	3
CanolaExtractor	4
content	4
DefaultExtractor	5
Extractor	6
KeepEverythingExtractor	7
LargestContentExtractor	7
NumWordsRulesExtractor	8

boilerpipeR-package *Extract the main content from HTML files*

Description

boilerpipeR interfaces the boilerpipe Java library, created by Christian Kohlschutter <http://code.google.com/p/boilerpipe/>. It implements robust heuristics to extract the main content from HTML files, removing unnessecary elements like ads, banners and headers/footers.

Author(s)

Mario Annau <mario.annau@gmail>

See Also

[Extractor](#) [DefaultExtractor](#) [ArticleExtractor](#)

Examples

```
## Not run:  
data(content)  
extract <- DefaultExtractor(content)  
cat(extract)  
  
## End(Not run)
```

ArticleExtractor *A full-text extractor which is tuned towards news articles.*

Description

In this scenario it achieves higher accuracy than [DefaultExtractor](#).

Usage

```
ArticleExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- ArticleExtractor(content)
```

ArticleSentencesExtractor

A full-text extractor which is tuned towards extracting sentences from news articles.

Description

A full-text extractor which is tuned towards extracting sentences from news articles.

Usage

```
ArticleSentencesExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- ArticleSentencesExtractor(content)
```

CanolaExtractor	<i>A full-text extractor trained on a Rhrefhttp://krdwr.org/krdwr Rhrefhttps://krdwr.org/trac/attachment/wiki/Corpora/Canola/CANOLA.pdfCanola.</i>
-----------------	--

Description

A full-text extractor trained on a [krdwr Canola](#).

Usage

```
CanolaExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- CanolaExtractor(content)
```

content	<i>Wordpress generated Webpage (retrieved from Quantivity Blog http://quantivity.wordpress.com). Content is saved as character and ready to be extracted.</i>
---------	---

Description

Wordpress generated Webpage (retrieved from Quantivity Blog <http://quantivity.wordpress.com>). Content is saved as character and ready to be extracted.

Author(s)

Mario Annau

References

<http://quantity.wordpress.com>

Examples

```
#Data set has been generated as follows:
## Not run:
library(RCurl)
url <- "http://quantity.wordpress.com/2012/11/09/multi-asset-market-regimes/"
content <- getURL(url)
content <- iconv(content, "UTF-8", "ASCII//TRANSLIT")
save(content, file = "content.rda")

## End(Not run)
```

DefaultExtractor *A quite generic full-text extractor.*

Description

A quite generic full-text extractor.

Usage

```
DefaultExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- DefaultExtractor(content)
```

 Extractor

Generic extraction function which calls boilerpipe extractors

Description

It is the actual workhorse which directly calls the boilerpipe Java library. Typically called through functions as listed for parameter exname.

Usage

```
Extractor(exname, content, asText = TRUE, ...)
```

Arguments

exname	character specifying the extractor to be used. It can take one of the following values: <ul style="list-style-type: none"> • ArticleExtractorA full-text extractor which is tuned towards news articles. • ArticleSentencesExtractorA full-text extractor which is tuned towards extracting sentences from news articles. • CanolaExtractorA full-text extractor trained on a krdwr. • DefaultExtractorA quite generic full-text extractor. • KeepEverythingExtractorMarks everything as content. • LargestContentExtractorA full-text extractor which extracts the largest text component of a page. • NumWordsRulesExtractorA quite generic full-text extractor solely based upon the number of words per block.
content	Text content or URL as character
asText	should content specified be treated as actual text to be extracted or url (from which HTML document is first downloaded and extracted afterwards), defaults to TRUE
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

References

<http://code.google.com/p/boilerpipe/>

KeepEverythingExtractor
Marks everything as content.

Description

Marks everything as content.

Usage

```
KeepEverythingExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- KeepEverythingExtractor(content)
```

LargestContentExtractor
A full-text extractor which extracts the largest text component of a page.

Description

For news articles, it may perform better than the [DefaultExtractor](#), but usually worse than [ArticleExtractor](#).

Usage

```
LargestContentExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- LargestContentExtractor(content)
```

NumWordsRulesExtractor

A quite generic full-text extractor solely based upon the number of words per block (the current, the previous and the next block).

Description

A quite generic full-text extractor solely based upon the number of words per block (the current, the previous and the next block).

Usage

```
NumWordsRulesExtractor(content, ...)
```

Arguments

content	Text content as character
...	additional parameters

Value

extracted text as character

Author(s)

Mario Annau

See Also

[Extractor](#)

Examples

```
data(content)
extract <- NumWordsRulesExtractor(content)
```

Index

*Topic **data**

content, [4](#)

*Topic **package**

boilerpipeR-package, [2](#)

ArticleExtractor, [2](#), [2](#), [6](#), [7](#)

ArticleSentencesExtractor, [3](#), [6](#)

boilerpipe (boilerpipeR-package), [2](#)

boilerpipeR-package, [2](#)

CanolaExtractor, [4](#), [6](#)

content, [4](#)

DefaultExtractor, [2](#), [5](#), [6](#), [7](#)

Extractor, [2–5](#), [6](#), [7–9](#)

KeepEverythingExtractor, [6](#), [7](#)

LargestContentExtractor, [6](#), [7](#)

NumWordsRulesExtractor, [6](#), [8](#)