Package ‘archiveRetriever’

June 21, 2022

Title  Retrieve Archived Web Pages from the 'Internet Archive'

Version  0.2.0

Description  Scraping content from archived web pages stored in
             the 'Internet Archive' (<https://archive.org>) using a systematic
             workflow. Get an overview of the mementos available from the
             respective homepage, retrieve the URLs and links of the page and
             finally scrape the content. The final output is stored in tibbles,
             which can be then easily used for further analysis.

License  Apache License (>= 2.0)

URL  https://github.com/liserman/archiveRetriever/

Imports  anytime, dplyr, ggplot2, gridExtra, httr, jsonlite, lubridate,
      rvest, stringr, tibble, tidyr, utils, xml2

Suggests  vcr (>= 0.6.0), testthat, webmockr

Encoding  UTF-8

RoxygenNote  7.2.0

NeedsCompilation  no

Author  Konstantin Gavras [aut] (<https://orcid.org/0000-0002-9222-0101>),
        Lukas Isermann [aut, cre] (<https://orcid.org/0000-0002-7195-9302>)

Maintainer  Lukas Isermann <lukas.isermann@uni-mannheim.de>

Repository  CRAN

Date/Publication  2022-06-21 15:20:05 UTC

R topics documented:

  archive_overview .................................................. 2
  retrieve_links ...................................................... 3
  retrieve_urls ...................................................... 3
  scrape_urls ....................................................... 4

Index  6
archive_overview

Description

archive_overview provides an overview of available mementos of the homepage from the Internet Archive.

Usage

archive_overview(homepage, startDate, endDate)

Arguments

homepage A character vector of the homepage, including the top-level-domain
startDate A character vector of the starting date of the overview. Accepts a large variety of date formats (see anytime)
endDate A character vector of the ending date of the overview. Accepts a large variety of date formats (see anytime)

Value

This function provides an overview of mementos available from the Internet Archive. It returns a calendar indicating all dates in which mementos of the homepage have been stored in the Internet Archive at least once. However, a memento being stored in the Internet Archive does not guarantee that the information from the homepage can be actually scraped. As the Internet Archive is an internet resource, it is always possible that a request fails due to connectivity problems. One easy and obvious solution is to re-try the function.

Examples

## Not run:
archive_overview(homepage = "www.spiegel.de", startDate = "20180601", endDate = "20190615")
archive_overview(homepage = "nytimes.com", startDate = "2018-06-01", endDate = "2019-05-01")

## End(Not run)
**retrieve_links**

*retrieve_links*: Retrieving Links of Lower-level web pages of mementos from the Internet Archive

**Description**

`retrieve_links` retrieves the Urls of mementos stored in the Internet Archive

**Usage**

`retrieve_links(ArchiveUrls, encoding = "UTF-8", ignoreErrors = FALSE)`

**Arguments**

- `ArchiveUrls`: A string of the memento of the Internet Archive
- `encoding`: Specify a encoding for the homepage. Default is 'UTF-8'
- `ignoreErrors`: Ignore errors for some Urls and proceed scraping

**Value**

This function retrieves the links of all lower-level web pages of mementos of a homepage available from the Internet Archive. It returns a tibble including the baseUrl and all links of lower-level web pages. However, a memento being stored in the Internet Archive does not guarantee that the information from the homepage can be actually scraped. As the Internet Archive is an internet resource, it is always possible that a request fails due to connectivity problems. One easy and obvious solution is to re-try the function.

**Examples**

```r
## Not run:
## End(Not run)
```

**retrieve_urls**

*retrieve_urls*: Retrieving Urls from the Internet Archive

**Description**

`retrieve_urls` retrieves the Urls of mementos stored in the Internet Archive

**Usage**

`retrieve_urls(homepage, startDate, endDate, collapseDate = TRUE)`

**Value**

This function retrieves the Urls of mementos stored in the Internet Archive. It returns a tibble with the Urls from the Internet Archive. The function can be used to filter the mementos by a specific date range. The `collapseDate` argument is used to specify how to handle dates in the mementos. If `collapseDate` is set to `TRUE`, the date range is collapsed into a single date. If not, the dates are kept as is. This is useful when the mementos were created on different dates but represent the same content. In such cases, it might be more useful to keep the dates.

**Examples**

```r
## Not run:
## End(Not run)
```
scrape_urls

Arguments

- **homepage**: A character vector of the homepage, including the top-level-domain.
- **startDate**: A character vector of the starting date of the overview. Accepts a large variety of date formats (see anytime).
- **endDate**: A character vector of the ending date of the overview. Accepts a large variety of date formats (see anytime).
- **collapseDate**: A logical value indicating whether the output should be limited to one memento per day.

Value

This function retrieves the mementos of a homepage available from the Internet Archive. It returns a vector of strings of all mementos stored in the Internet Archive in the respective time frame. The mementos only refer to the homepage being retrieved and not its lower level web pages. However, a memento being stored in the Internet Archive does not guarantee that the information from the homepage can be actually scraped. As the Internet Archive is an internet resource, it is always possible that a request fails due to connectivity problems. One easy and obvious solution is to re-try the function.

Examples

```r
## Not run:
retrieve_urls("www.spiegel.de", "20190801", "20190901")
retrieve_urls("nytimes.com", startDate = "2018-01-01", endDate = "01/02/2018")
retrieve_urls("nytimes.com", startDate = "2018-01-01", endDate = "2018-01-02", collapseDate = FALSE)
## End(Not run)
```

scrape_urls

**scrape_urls**: ScrapingUrls from the Internet Archive

Description

**scrape_urls** scrapes URLs of mementos and lower-level web pages stored in the Internet Archive using XPaths as default.

Usage

```r
scrape_urls(
  Urls,
  Paths,
  collapse = TRUE,
  startnum = 1,
  attachto = NULL,
  CSS = FALSE,
  archiveDate = FALSE,
)```
scrape_urls

ignoreErrors = FALSE,
stopatempty = TRUE,
emptylim = 10,
encoding = "UTF-8",
lengthwarning = TRUE
)

Arguments

Urls A character vector of the memento of the Internet Archive
Paths A named character vector of the content to be scraped from the memento. Takes
XPath expressions as default.
collapse Collapse matching html nodes
startnum Specify the starting number for scraping the Urls. Important when scraping
breaks during process.
attachto Scraper attaches new content to existing object in working memory. Object
should stem from same scraping process.
CSS Use CSS selectors as input for the Paths
archiveDate Retrieve the archiving date
ignoreErrors Ignore errors for some Urls and proceed scraping
stopatempty Stop if scraping does not succeed
emptylim Specify the number of Urls not being scraped until break-off
encoding Specify a default encoding for the homepage. Default is 'UTF-8'
lengthwarning Warning function for large number of URLs appears. Set FALSE to disable
default warning.

Value

This function scrapes the content of mementos or lower-level web pages from the Internet Archive.
It returns a tibble including Urls and the scraped content. However, a memento being stored in
the Internet Archive does not guarantee that the information from the homepage can be actually
scraped. As the Internet Archive is an internet resource, it is always possible that a request fails due
to connectivity problems. One easy and obvious solution is to re-try the function.

Examples

## Not run:
sanitize_urls(
Paths = c(title = "//article/div/h2//text()", teaser = "//article/div/p/text()")
)

## End(Not run)
Index

anytime, 2, 4
archive_overview, 2

retrieve_links, 3
retrieve_urls, 3

scrape_urls, 4