Nexis UK scraper

Kohei Watanabe

October 25, 2017

Contents

The Nexis scraper in R

This is an R package to automatically search and download news articles from the Nexis database. This package is developed aiming to promote large scale content analysis that investigates problems in mass communication in information-saturated society.

LexisNexis demands users high extra fees to access to the Nexis API, preventing the academics from downloading and analysing large dataset in research. Until the company to decide offering academic users the API for no or small extra costs, scraping is the only way for them to embark on large scale analysis of news content.

This package provides automated access to the Nexis database through a web browser. Web browsing is manipulated using **Selenium** to search and download news articles without human attendance. Depending on the Nexis database's response time, this scraper can download approximately 30,000 news articles per hour.

Target database

The Nexis database has different interfaces for users in different countries. This scraper is designed for Nexis UK, which is used in the UK and Irish universities. For other versions of the Nexis database, functions in this package needs to be modified.

Supported browser

Currently, this package only supports scraping by **Firefox**, but it is relatively easy to add **Chrome**.

Future updates

Web scraping is not a reliable technology as it depends on the HTML tags in web pages that are frequently updated. Therefore, this scraper has to be well maintained to keep up with the changes in Nexis database. If the scraper suddenly stops working, try the latest version available in this repository. If it does not solve the problem, please file an issue or writer a patch and submit a pull request.

Redistributon

This package is open-source program license under GPL-3, so you can share it with your friends. If your friends wish to use the latest version of the package, contact Kohei Watanabe (watanabe.kohei@gmail.com) requesting access to the repository. However, please do not to upload the package to a website anyone can access to, because scraping is not officially allowed by LexisNexis.

How to use

System setup

Install JRE

You have to install Java Runtime Environment 8 to use this package. JRE is required to run the Selenium server. The JRE 9 is the latest, but it tend to have problem with Selenium.

Start Selenium server

You have to run the Selenium sever *before* using this package. Selenium Standalone Server is available at SeleniumHQ. This repository also contains Selenium Standalone Sever (selenium-server-standalone-3.4.0.jar) in a sub-directory for download, but not a web driver for Firefox (**geckodriver**). Please download **geckodriver** and save it to the same directory as the Selenium server.

To start Selenium Standalone Sever, you only need to run the following command in the console:

```
java -jar selenium-server-standalone-3.4.0.jar
```

Sometimes, you have to tell **Selenium** the location of **geckodriver** (in the same directory, in this example):

```
java -jar -Dwebdriver.gecko.driver=./geckodriver selenium-server-standalone-3.4.0.jar
```

On Windows, the command looks slightly different:

```
java.exe -Dwebdriver.gecko.driver=./geckodriver.exe -jar selenium-server-standalone-3.4.0.jar
```

Note that you have the directory that contains Java executable in the system path. On Windows, `java.exe` is usually located in `C:\Program Files (x86)\Java\jre1.8.0_144\bin`. Please refer to other source for how to add a directory to the system path. After adding to the system path, your Windows needs restart.

You can stop the Selenium server by pressing `Ctrl + C`.

Package setup

Install

Since this package is in a private repository, you cannot use `devtools::install_github('koheiw/Nexis')`. You have to either clone the repository and build or download in a zip file and install:

```
devtools::install("/home/kohei/Downloads/Nexis-master/") # unzipped folder
require('Nexis')
```

File location

You first have to set a directory where downloaded files are saved:

```
set_directory('/home/kohei/Documents/Nexis')
```

Login to database

You have to login to the Nexis database using your library account in the browser windows opened up by `open_browser()`. You can manually navigate to the database from the library website, or can directly access from https://www.nexis.com using *Academic Sign-in*:

```
url_login <- 'https://www.lexisnexis.com/start/shib/idpurlrd?entityID=https%3A%2F%2Flse.ac.uk%2Fidp&requ

# or

#url_login <- 'http://www.lse.ac.uk/library'

open_browser(url_login) # a new browser window will open
#driver <- get_driver() # backup browser connection (for development only)
```

Download setting

Before starting download, you have to decide search query, download period, and size of search window. In this example, the scraper will download news articles that contain "Brexit" published between 1 January 2016 and 31 December 2016 separately for each month:

```
query <- "Brexit"
from <- '2016-01-01'
to <- '2016-12-31'
size <- 1
unit <- 'month'
```

Start download

Finally, start automated download:

```
#set_driver(driver) # restore browser connection (for development only)
check_login()
dates <- get_date_range(from, to, size, unit)
for (i in seq_along(dates)) {
    if (is_completed(dates[[i]])) next
    submit(query, dates[[i]]) # please check if the default date format ("%m/%d/%Y") is currect
    if (is_zero(dates[[i]])) next
    ranges <- get_download_range(size = 500) # download 500 items each time
    for (j in seq_along(ranges)) {
        download(dates[[i]], ranges[[j]], tail(unlist(ranges), 1))
    }
}
```

Extract texts from downloaded files

Once download has been completed, you can extract texts from the downloaded HTML files using `import_nexis()`.

```
data <- import_nexis('/home/kohei/Documents/Nexis')
head(data)
```