# Week 9: Web APIs

Alina Vikhnevich

2025-03-31

## Introduction

For this assignment, I explored how to connect to a RESTful API and extract structured data from the web. I used the *New York Times Most Popular API*, which provides access to a feed of the most-emailed articles over the past week. The overall objective was to pull this data in JSON format, parse it, clean it, and finally transform it into a tidy R `data.frame`, which I then saved as an Excel file for future use.

This exercise gave me hands-on experience with API calls using `httr2`, working with JSON responses, and cleaning nested data structures skills that are crucial for web-based data acquisition and preparation in real-world data science workflows.

### Load Required Libraries

```r
# Load required libraries
library(httr2) # For making HTTP requests
library(jsonlite) # For parsing JSON
library(tidyverse) # For data manipulation and cleaning
library(writexl) # To save the result
```

I started by loading all the libraries needed to complete the task. `httr2` is used to perform the HTTP request to the API. `jsonlite` helps parse the JSON response into a format that R can work with. `tidyverse` is essential for data wrangling, and `writexl` lets me export the cleaned dataset to Excel for easier review and sharing.

### 1. Store API Key and Send the Request

```r
# Store your API key
api_key <- Sys.getenv("NYT_API_KEY")

# Create and send the request
resp <- request("https://api.nytimes.com/svc/mostpopular/v2/emailed/7.json") %>%
  req_url_query("api-key" = api_key) %>%
  req_perform()
```

I registered for an API key from the NYT Developer Portal. To securely manage access credentials, I stored my New York Times API key in the `.Renviron` file and retrieved it in the script using `Sys.getenv("NYT_API_KEY")`. It keeps the key hidden from the rendered document. Then, I constructed the API request using `httr2`, added the key as a query parameter, and performed the request. This returned a live HTTP response containing JSON data of the top-emailed articles.

## 2. Parse JSON Content

```
# Parse the JSON content
resp_text <- resp_body_string(resp)
data_parsed <- fromJSON(resp_text, flatten = TRUE)
```

The raw JSON content from the API response was first converted into a character string, and then parsed using `fromJSON()`. I used `flatten = TRUE` so that any nested data structures were simplified into a flat data frame format. This helps avoid complex list-columns later in the analysis.

## 3. Extract the Articles Section

```
# Extract just the 'results' section
articles_df <- data_parsed$results

# Take a quick look
glimpse(articles_df)
```

```
## Rows: 20
## Columns: 22
## $ uri            <chr> "nyt://article/f0e4153a-1d52-5ab6-b3e4-457ce1512127", "~
## $ url            <chr> "https://www.nytimes.com/2025/03/26/world/asia/south-ko~
## $ id             <dbl> 1e+14, 1e+14, 1e+14, 1e+14, 1e+14, 1e+14, 1e+14, 1e+14,~
## $ asset_id       <dbl> 1e+14, 1e+14, 1e+14, 1e+14, 1e+14, 1e+14, 1e+14, 1e+14,~
## $ source         <chr> "New York Times", "New York Times", "New York Times", "~
## $ published_date <chr> "2025-03-26", "2025-03-24", "2025-03-26", "2025-03-28",~
## $ updated        <chr> "2025-03-27 05:45:18", "2025-03-27 03:09:59", "2025-03-~
## $ section        <chr> "World", "Well", "Well", "Well", "Travel", "Opinion", "~
## $ subsection     <chr> "Asia Pacific", "Move", "Move", "", "", "", "", "", "",~
## $ nytdsection    <chr> "world", "well", "well", "well", "travel", "opinion", "~
## $ adx_keywords   <chr> "Adoptions;Corruption (Institutional);Politics and Gove~
## $ column         <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ byline         <chr> "By Choe Sang-Hun", "By Christine Yu and Theodore Tae",~
## $ type           <chr> "Article", "Article", "Article", "Article", "Interactiv~
## $ title          <chr> "World’s Largest ‘Baby Exporter’ Admits to Adoption Fra~
## $ abstract       <chr> "A South Korean truth commission called for the country~
## $ des_facet      <list> <"Adoptions", "Corruption (Institutional)", "Politics ~
## $ org_facet      <list> <>, <>, <>, <>, <"Four Seasons Hotels Ltd", "United Na~
## $ per_facet      <list> <>, <>, <>, <>, "Capa, Robert", "Trump, Donald J", <"F~
## $ geo_facet      <list> "South Korea", <>, <>, <>, "Budapest (Hungary)", <>, "~
## $ media          <list> [<data.frame[1 x 6]>], [<data.frame[1 x 6]>], [<data.f~
## $ eta_id         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

The API response includes metadata along with the actual content we’re interested in. I isolated the `results` list, which contains the articles themselves, and assigned it to a new data frame.

## 4. Clean and Tidy the Dataset

```
# Clean and structure the data
clean_articles <- articles_df %>%
  select(
    title,
    byline,
    section,
    published_date,
    source,
    abstract,
    url
  ) %>%
  mutate(across(everything(), ~ ifelse(. == "" | is.na(.), "None", .)))

# Preview the cleaned dataset
head(clean_articles)
```

```
##                                                          title
## 1    World’s Largest ‘Baby Exporter’ Admits to Adoption Fraud
## 2           A 10-Minute Evening Yoga Routine for Better Sleep
## 3                         Knees Hurt? This Exercise Can Help.
## 4 The Worst Habits for Your Back, According to Spine Surgeons
## 5                                        36 Hours in Budapest
## 6                      The Democrats Are in Denial About 2024
##                              byline section published_date         source
## 1                  By Choe Sang-Hun   World     2025-03-26 New York Times
## 2 By Christine Yu and Theodore Tae    Well     2025-03-24 New York Times
## 3                 By Hilary Achauer    Well     2025-03-26 New York Times
## 4                    By Jancee Dunn    Well     2025-03-28 New York Times
## 5                      By Evan Rail  Travel     2025-03-27 New York Times
## 6            By The Editorial Board Opinion     2025-03-29 New York Times
##
## 1 A South Korean truth commission called for the country to apologize to those who were sent abroad
## 2                                                                                         These begi
## 3                                                  Strength training can be one of the best treatmen
## 4
## 5                         New museums, galleries and spruced-up parks counterbalance this Central Eu
## 6                               Party leaders have embraced convenient excuses. This perilous politi
##                                                                                 url
## 1    https://www.nytimes.com/2025/03/26/world/asia/south-korea-adoption-fraud.html
## 2           https://www.nytimes.com/2025/03/24/well/move/evening-yoga-routine.html
## 3    https://www.nytimes.com/2025/03/26/well/move/strength-training-knee-pain.html
## 4      https://www.nytimes.com/2025/03/28/well/bad-habits-back-spine-surgeons.html
## 5 https://www.nytimes.com/interactive/2025/03/27/travel/things-to-do-Budapest.html
## 6          https://www.nytimes.com/2025/03/29/opinion/democrats-strategy-2024.html
```

The raw dataset contained many columns, some of which were either nested or not useful for my purposes. I kept only the key columns needed for a readable summary: title, author, section, publication date, source, summary, and URL. To handle missing `byline` entries, I replaced empty or `NA` values with `"No Author"`, which keeps the dataset more readable.

## 5. Save the Results to Excel

```r
# Save cleaned version to Excel file
write_xlsx(clean_articles, "nyt_popular_articles.xlsx")
```

Once the data was cleaned, I exported it to an Excel file. I chose .xlsx instead of .csv to avoid encoding issues with special characters (e.g., apostrophes and quotation marks) that showed up when exporting to CSV. This method ensured all article titles and abstracts remained readable and intact.

## Conclusion

Working through this assignment gave me a clearer understanding of how to source data programmatically using a web API. The process began with creating and sending an authenticated request using my API key via the `httr2` package. I found `httr2` to be very intuitive once I got the structure of the request right.

Parsing the JSON response into an R-readable format using `jsonlite::fromJSON()` with `flatten = TRUE` was a critical step that helped reduce nested complexity early on. After extracting just the `results` portion of the response, I focused on selecting and cleaning the relevant columns. This included handling missing author names in the `byline` column and preparing the dataset for analysis or export.

One minor challenge I ran into was character encoding when attempting to save the dataset as a `.csv` - certain special characters in article titles and abstracts didn't render properly. Switching to an `.xlsx` format using `writexl::write_xlsx()` solved the issue and preserved the formatting, ensuring the data remained clean and readable.