

# Hw 12

Leah Fawzi

12/4/2018

```
library(httr)
```

I chose the newspaper CNN en Español, the Spanish version of CNN, and created 2 functions that search the newspaper and interact with the newspaper data.

First, I created a function, `get_articles`, that finds the titles of 5 recent articles for a specific topic.

```
get_articles <- function(topic, api_key){
  base <- "https://newsapi.org/v2/everything?sources=cnn-es"

  base2 <- "q="
  end_base <- "apiKey="

  url <- paste(base, "&", base2, topic, "&", end_base, api_key, sep="")

  x <- httr::GET(url=url)
  x <- jsonlite::fromJSON(rawToChar(x$content))
  titles <- x$articles$title[1:5]
  return(titles)
}
```

Here is an example of getting the titles of 5 recent articles regarding terrorism:

```
get_articles("terrorismo", "4cc76b2120ca48169b5b0466236a98c3")
```

```
## [1] "EE.UU. debate incluir a Venezuela en la lista de países patrocinadores del terrorismo"
## [2] "Ataque en Melbourne deja al menos un muerto y dos heridos"
## [3] "Pareja que nombró a su bebé Adolf por Hitler fueron hallados culpables de ser miembros de grupo"
## [4] "Cuatro ataques en tan solo dos semanas en EE.UU.: ¿cómo proteger estos blancos fáciles frecuent"
## [5] "¿Quién reemplazará al secretario de Justicia Jeff Sessions? Ocho posibles candidatos"
```

Next, I created a function, `get_headlines`, that finds the titles of headlines from a specific time period.

```
get_headlines <- function(api_key){
  base <- "https://newsapi.org/v2/top-headlines?sources=cnn-es"
  end_base <- "apiKey="

  url <- paste(base, "&", end_base, api_key, sep="")
  x <- httr::GET(url=url)
  x <- jsonlite::fromJSON(rawToChar(x$content))
  titles <- x$articles$title
  return(titles)
}
```

Here is an example of getting the titles of headlines.

```
get_headlines("4cc76b2120ca48169b5b0466236a98c3")
```

```
## [1] "Latinoamérica, la región más violenta del planeta: 5 ciudades mexicanas en el top 10"
## [2] "Encuentran un cuerpo cerca del último lugar en Costa Rica donde se cree que estuvo una ciudadana"
## [3] "EN VIVO: Los funerales de George Bush padre"
```

```
## [4] "5 cosas que debes saber este 4 de diciembre: el legado de Bush, el miedo de Khashoggi, el verda
## [5] "Lenín Moreno habló con Manafort sobre remover a Assange de la embajada de Ecuador en Londres"
## [6] "Ella quería ayudar a una madre necesitada y a su bebé, pero murió acuchillada"
```

Then, I created a function to translate any text into English.

```
translate_text <- function(google_api_key, source, target, text){
  body <- paste("{",
                "'q':", text, ",",
                "'source':", source, ",",
                "'target':", target, ",",
                "'format':'text'",
                "}", sep="")

  base <- "https://translation.googleapis.com/language/translate/v2?key="
  url <- paste(base, google_api_key, sep="")
  out <- rep(NA, length(body))
  for(i in 1:length(body)){
    y <- httr::POST(url, body = body[i])
    y <- jsonlite::fromJSON(rawToChar(y$content))
    out[i] <- y$data$translations$translatedText
  }
  return(out)
}
```

As an example, I will translate the titles of the articles that I produced with the first function.

```
translate_text("AIzaSyB-iMKY9sQ2gTPvrD0btoD7U2rZBwr3fas", "es", "en", get_articles("terrorismo", "4cc76b2
## [1] "USA debate to include Venezuela in the list of sponsoring countries of terrorism"
## [2] "Attack in Melbourne leaves at least one dead and two injured"
## [3] "Couple who named their baby Adolf by Hitler were found guilty of being members of a banned neo-f
## [4] "Four attacks in just two weeks in the US: how to protect these easy targets frequented by peopl
## [5] "Who will replace Attorney General Jeff Sessions? Eight possible candidates"
```

Finally, I will convert text to speech with a function. This function creates a text file of the sound and an audio file of the translated text.

```
text_to_speech <- function(text, google_api_key){
  text2 <- paste(text, collapse=" ")

  input <- paste("'input':{'text':", text2, "}", sep="")
  voice <- "'voice':{'languageCode':'en-US','name':'en-US-Standard-C','ssmlGender':'FEMALE'}"
  output <- "'audioConfig':{'audioEncoding':'MP3'}"

  s_body <- paste("{", input, ",", voice, ",", output, "}", sep="")

  base_url <- "https://texttospeech.googleapis.com/v1/text:synthesize?key="
  url <- paste(base_url, google_api_key, sep="")
  x <- httr::POST(url, body = s_body)
  x <- jsonlite::fromJSON(rawToChar(x$content))

  Sys.setenv(output_string=x$audioContent)
  system("echo $output_string > sound_out.txt")

  system("base64 --decode sound_out.txt > sound_out.mp3")
  system("afplay sound_out.mp3")
}
```

```
}
```

For example, I convert the translated text from above to an audio file with the code below.

```
text_to_speech(translate_text("AIzaSyB-iMKY9sQ2gTPvrD0btoD7U2rZBwr3fas", "es","en", get_articles("terror
```