R Script IMDB web

Hanna Asipovich

2022-12-11

rm(list=ls())

```
#Loading the rvest package
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.1.3
```

```
#Get top 250 movies on IMDB

url <- 'https://www.imdb.com/chart/top'
# read url
file_html <- read_html(url)
# write_html(file_html, 'html_file.html')
# assign data from titleColumn for reusability
boxes <- file_html %>%
  html_nodes('.titleColumn a')
# get movie names
movie_names <- html_text(boxes)
# get movie ratings
ratings <- as.numeric(
  file_html %>%
    html_nodes('strong') %>%
    html_text())
# get movie released year
years_p <-
  file_html %>%
  html_nodes('.secondaryInfo') %>%
  html_text()
# remove brackets ()
years <- as.numeric(sub("\\).*", "", sub(".*\\(", "", years_p)))
# get link of the movie
links_p <-
  boxes %>%
  html_attr('href')

# get actors of the movie
actors <-
  boxes %>%
  html_attr('title')
# get votes
votes_p <- file_html %>%
```

```
  html_nodes('.imdbRating') %>%
  html_node('strong') %>%
  html_attr('title')

# extract only numbers from votes as string
votes_s <- c()
for (x in votes_p) {
  votes_s <- c(votes_s, gsub(',','',
                             gsub(' user ratings','',
                                  gsub('.*?based on ','',
                                       x))))
}
votes <- c()
# convert votes string into numeric
for (x in votes_s) {
  votes <- c(votes, as.numeric(x))
}
```

```
#Get bottom 100 movies on IMDB
url2 <- 'https://www.imdb.com/chart/bottom'
# read url
file_html2 <- read_html(url2)
# write_html(file_html, 'html_file.html')
# assign data from titleColumn for reusability
boxes2 <- file_html2 %>%
  html_nodes('.titleColumn a')
# get movie names
movie_names2 <- html_text(boxes2)
# get movie ratings
ratings2 <- as.numeric(
  file_html2 %>%
    html_nodes('strong') %>%
    html_text())
# get movie released year
years_p2 <-
  file_html2 %>%
  html_nodes('.secondaryInfo') %>%
  html_text()
# remove brackets ()
years2 <- as.numeric(sub("\\).*", "", sub(".*\\(", "", years_p2)))
# get link of the movie
links_p2 <-
  boxes2 %>%
  html_attr('href')

# get actors of the movie
actors2 <-
  boxes2 %>%
  html_attr('title')
# get votes
votes_p2 <- file_html2 %>%
  html_nodes('.imdbRating') %>%
  html_node('strong') %>%
```

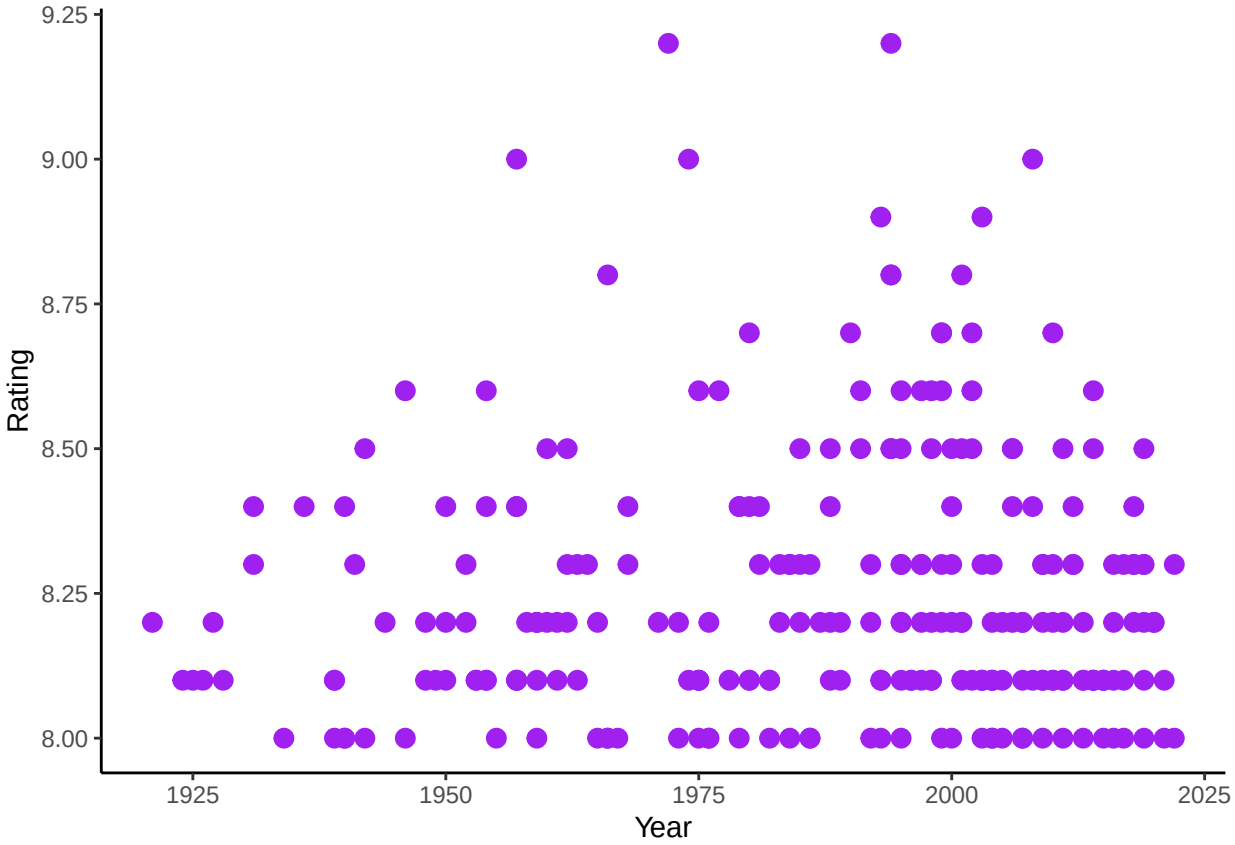
```
  html_attr('title')

# extract only numbers (as a string) from full string
votes_s2 <- c()
for (x in votes_p2) {
  votes_s2 <- c(votes_s2, gsub(',','',
                               gsub(' user ratings','',
                                    gsub('.*?based on ','',
                                         x))))
}
votes2 <- c()
# convert string into numeric
for (x in votes_s2) {
  votes2 <- c(votes2, as.numeric(x))
}
```

```
# create a dataframe Top 250 movies for further analysis. votes as numeric
df <- data.frame('movie_title' = movie_names,
                 'year' = years,
                 'rating' = ratings,
                 'cast' = actors,
                 'votes'=votes)
```

```
# create a dataframe for bottom movies
df2 <- data.frame('movie_title2' = movie_names2,
                          'year2' = years_p2,
                          'rating2' = ratings2,
                          'cast2' = actors2,
                         'votes2'=votes2)
```

```
#ggplot for top 250  movies
ggplot(data=df)+
  geom_point(aes(x=years, y=ratings, color=votes), size=3, color= "purple")+
  labs(x = 'Year',
       y = 'Rating')+
  scale_colour_manual(values = colors) +
  theme_classic()
```




```
#ggplot for worst movies
ggplot(data=df2)+
  geom_point(aes(x=years2, y=ratings2, color=votes2), size=3, color="pink")+
  labs(x = 'Year',
       y = 'Rating')+
  theme_classic()
```


```
library("aws.comprehend")
```

```
detect_language("Andy Dufresne: [referring to Andy using an alias to launder money for the warden] If t
Andy Dufresne: Yeah. The funny thing is - on the outside, I was an honest man, straight as an arrow. I
```

```
##   Index LanguageCode     Score
## 1     0           en 0.9957539
```

Detect sentiment in piece used in Perspective API

```
detect_sentiment("HADLEY
        YOU EAT WHEN WE SAY YOU EAT! YOU
        PISS WHEN WE SAY YOU PISS! YOU SHIT
        WHEN WE SAY YOU SHIT! YOU SLEEP
        WHEN WE SAY YOU SLEEP! YOU MAGGOT-
        DICK MOTHERFUCKER!")
```

```
##   Index Sentiment       Mixed Negative    Neutral    Positive
## 1     0  NEGATIVE 2.40673e-05 0.969276 0.02206963 0.008630243
```

Detect sentiment in transcribed piece

```
detect_sentiment("Ladies and gentlemen, you've heard all the evidence. I submit that this was not a hot
```

```
##   Index Sentiment     Mixed   Negative  Neutral  Positive
## 1     0  POSITIVE 0.3285505 0.06327507 0.189357 0.4188174
```