LabExercise5

Ann Margaret Camayodo

2024-03-17

cleaning data of my Lab Exercise #4

```
library(readr)
library(stringr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
arxiv <- read_csv("myCSV_files/Arxiv_shoes.csv")
```

```
## New names:
## * `` -> `...1`
```

```
## Rows: 150 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (5): title, author, subject, abstract, meta
## dbl (1): ...1
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
arxiv_date_only <- str_extract(arxiv$meta, "\\d+\\s[A-Za-z]+\\s\\d+")

arxiv_date_type <- as.Date(arxiv_date_only, format = "%d %b %Y")
head(arxiv_date_type)
```

```
## [1] "2024-02-05" "2024-01-03" "2024-01-16" "2024-01-13" "2024-01-11"
## [6] "2023-12-29"
```

```
cleaned_arxiv <- arxiv %>%
  mutate(date = arxiv_date_type) %>%
  mutate(subject = gsub("\\s\\(.*\\)", "", subject),
         across(where(is.character), tolower)) %>%
  select(-meta, -...1)
```

```r
write.csv(cleaned_arxiv, "cleanedData/Cleaned_arvixPaperSHOES.csv")
```

cleaning data of my Lab Exercise #2

```r
library(readr)
library(stringr)
library(dplyr)


movieReviews <- read_csv("myCSV_files/movieReviews.csv")
```

```
## New names:
## Rows: 2450 Columns: 7
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (6): movie_name, title, reviewer, review, date, ratings dbl (1): ...1
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

```r
reviews_date_type <- as.Date(str_extract(movieReviews$date, "\\d+\\s[A-Za-z]+\\s\\d+"), format = "%d %b

reviews_ratings_integer <- as.integer(str_extract(movieReviews$ratings, "\\d+\\.\\d+"))

# removing emoji/emoticons
movieReviews$title <- gsub("\\p{So}", "", movieReviews$title, perl = TRUE)
movieReviews$reviewer <- gsub("\\p{So}", "", movieReviews$reviewer, perl = TRUE)
movieReviews$review <- gsub("\\p{So}", "", movieReviews$review, perl = TRUE)

#removing non-English language
movieReviews$title <- gsub("[^a-zA-Z ]", "", movieReviews$title)
movieReviews$reviewer <- gsub("[^a-zA-Z ]", "", movieReviews$reviewer)
movieReviews$review <- gsub("[^a-zA-Z ]", "", movieReviews$review)

#replacing the blanks for NA
movieReviews$title <- na_if(movieReviews$title, "")
movieReviews$reviewer <- na_if(movieReviews$reviewer, "")
movieReviews$review <- na_if(movieReviews$review, "")

#converting to lowercase...
movieReviews <- movieReviews %>%
  mutate(across(where(is.character), tolower)) %>%
  select(-...1)

# combining...
cleaned_reviews <- movieReviews %>%
  mutate(date = reviews_date_type, ratings = reviews_ratings_integer)

# writing as CSV
write.csv(cleaned_reviews, "cleanedData/Cleaned_movieReviews.csv")
```