Topic Modeling

Gary Wang

2024-11-01

```r
library(lexicon)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(topicmodels)
library(tidytext)
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```r
library(dplyr)
```

```r
movies <- read.csv("movie_plots.csv")
```

Group by Genre, summarize common words, and find genre frequency

```r
# Extract genres based on common keywords found in the plot descriptions
movies <- movies %>%
  mutate(Genre = case_when(
    str_detect(Plot, "(?i)science|space|experiment|future|alien|
               robot|technology|planet|human|new world|earth") ~ "Sci-Fi",
    str_detect(Plot, "(?i)love|romance|relationship|affair|
               wedding|couple|meets|girl") ~ "Romance",
    str_detect(Plot, "(?i)war|battle|army|soldier|conflict|
               military|enemy|tank|battlefield|dead") ~ "War",
    str_detect(Plot, "(?i)ghost|haunt|horror|fear|terror|
               scary|supernatural|creepy") ~ "Horror",
    str_detect(Plot, "(?i)crime|detective|murder|investigate|
               thriller|mafia|heist|mystery") ~ "Crime",
    str_detect(Plot, "(?i)action|fight|fighting|adventure|hero|explosion|
               battle|rescue") ~ "Action",
    str_detect(Plot, "(?i)comedy|funny|humor|laugh|joke|
               satire|parody") ~ "Comedy",
    str_detect(Plot, "(?i)history|historical|biography|true story|
               period drama|century|ancient") ~ "History",
    str_detect(Plot, "(?i)fantasy|magic|myth|legend|superhero|
               kingdom|evil") ~ "Fantasy",
    str_detect(Plot, "(?i)western|cowboy|wild west|sheriff|ranch|
               town|outlaw") ~ "Western",
    str_detect(Plot, "(?i)documentary|docu|true events|reality|
               biopic") ~ "Documentary",
    str_detect(Plot, "(?i)sport|game|team|match|championship|
               wrestling") ~ "Sport",
    str_detect(Plot, "(?i)home|people|brother|daughter|brothers|
               friend|wife|son|father|mother") ~ "Family",
    TRUE ~ "Other"
  ))

# Calculate the frequency of each genre
genre_frequency <- movies %>%
  count(Genre, name = "Frequency")

# Tokenize the plots and remove stop words
plot_words <- movies %>%
  unnest_tokens(word, Plot) %>%
  anti_join(get_stopwords()) %>%
  count(Genre, word, sort = TRUE)
```

```
## Joining with ‘by = join_by(word)‘
```

```r
# Group by Genre, summarize common words, and find genre frequency
nested_data <- plot_words %>%
  group_by(Genre) %>%
  summarize(
    Words = paste(unique(word), collapse = ", ")
  ) %>%
  left_join(genre_frequency, by = "Genre")

view(nested_data)
```