[Stat 184] Final Project

Netflix Content Analyis

Allan Samoilovich Junghyeon Sung Taegwon Lee

2024-12-18

1 Introdcution

"Netflix" is the most popular media platform these days. It would not be an exaggeration to call it a pioneer of trends. Netflix has revolutionized the way people consume entertainment by introducing binge-watching, personalized recommendations, and producing high-quality original content. Its innovative approach to streaming has set a new standard for the industry, influencing how competitors deliver and create content. As media consumption continues to rise, we chose a netflix dataset to explore Netflix's trends.

1.1 Dataset

We found our dataset from Kaggle, a well-known platform for sharing datasets and conducting data-driven analysis. The dataset, titled "Netflix Movies and TV Shows" [1], was collected by Shivam Bansal. It contains metadata about Netflix's catalog, including movies and TV shows, and was created to analyze trends, content distribution, and patterns in Netflix's offerings. Each row in the dataset corresponds to a specific piece of content, such as a movie or TV show, and includes attributes like title, director, cast, country, release year, duration, date added, and genre.

The dataset adheres to the FAIR principles. It is findable because it has a clear and descriptive title and file name (netflix_titles.csv) that allow researchers to locate it easily. It is accessible as it is publicly available for download on Kaggle. The data is interpretable because it uses standardized formats and clear column names such as title, type, and country, which are intuitive and require no additional documentation. Finally, the dataset ensures reusability by including detailed metadata and a clear structure, allowing it to be reused for various types of research and analysis.

[1] Bansal, S. (2023). *Netflix Movies and TV Shows Dataset.* https://www.kaggle.com/datasets/shivamb/netflix-shows

For our analysis, we focused on several key attributes to address research questions and explore trends within the Netflix catalog. These attributes include type, which differentiates between movies and TV shows; duration, which provides the length of movies in minutes or the number of seasons for TV shows; and date_added, which indicates when content was added to Netflix, allowing us to analyze trends in yearly additions. We also examined the listed_in column to identify popular genres, the country column to explore regional content contributions, and the cast column to determine the most frequently featured actors or actresses. Additionally, we analyzed the description column to identify recurring themes and keywords in Netflix's content.

All figures and tables in this report include descriptive captions and are cross-referenced within the text to maintain clarity and ensure that the results are easy to follow. These visualizations and summaries first highlight key insights, such as the distribution of content types, yearly content trends, top genres, top actors and the countries contributing most to Netflix's library. By using this background information, we collected 3 research questions and findings.

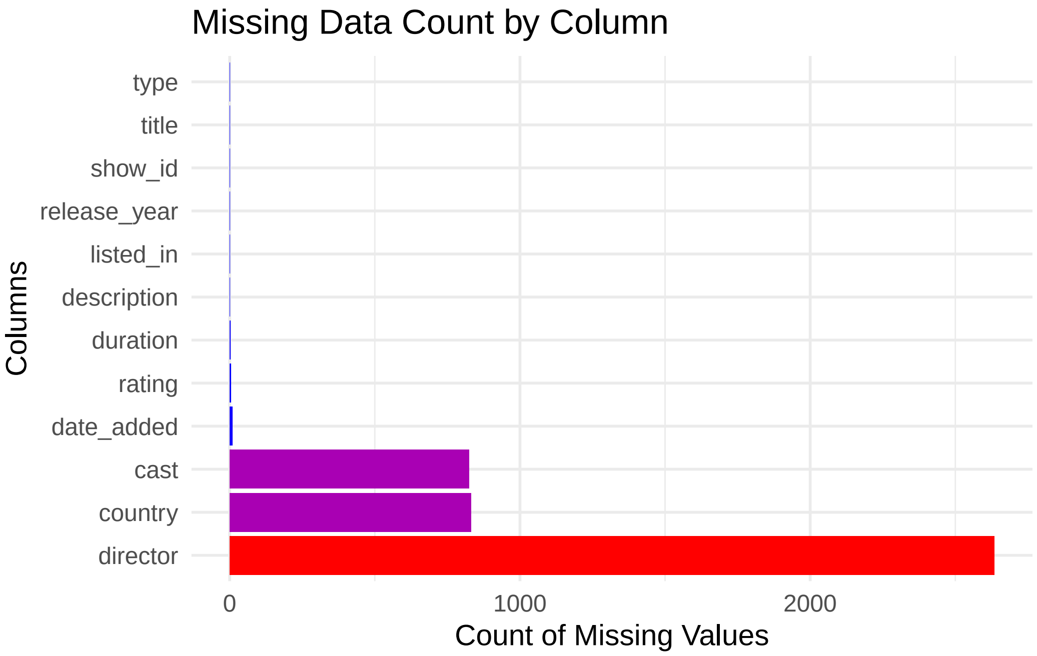
1.2 Data Wrangling Process

1.2.1 Check the entire dataset

Table 1: Descriptive Statistics for Entire Dataset

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
show_id*	1	8807	4404.00	2542.51	4404	4404.00	3264.69	1	8807	8806	0.00	-1.20	27.09
type*	2	8807	1.30	0.46	1	1.25	0.00	1	2	1	0.85	-1.27	0.00
title*	3	8807	4404.00	2542.51	4404	4404.00	3264.69	1	8807	8806	0.00	-1.20	27.09
director*	4	8807	1625.84	1525.04	1361	1504.78	2016.34	1	4529	4528	0.38	-1.31	16.25
cast*	5	8807	3487.46	2393.87	3435	3450.17	3144.59	1	7693	7692	0.07	-1.26	25.51
country*	6	8807	388.52	222.56	437	406.00	249.08	1	749	748	-0.44	-1.19	2.37
date_added*	7	8807	899.53	497.90	914	901.31	650.86	1	1768	1767	-0.03	-1.20	5.31
release_year	8	8807	2014.18	8.82	2017	2016.03	2.97	1925	2021	96	-3.45	16.22	0.09
rating*	9	8807	12.01	1.97	13	12.09	2.97	1	18	17	-0.48	0.73	0.02
duration*	10	8807	95.72	88.18	56	92.14	80.06	1	221	220	0.26	-1.69	0.94
listed_in*	11	8807	273.40	131.06	290	278.41	131.95	1	514	513	-0.31	-0.76	1.40
description*	12	8807	4386.75	2532.81	4386	4386.36	3249.86	1	8775	8774	0.00	-1.20	26.99

1.2.2 Handling Missing Values



- This graph shows that there are a lot of missing values in ‘director’ column.

1.2.3 Checked the cleaned dataset

Table 2: Descriptive Statistics for Cleaned Dataset

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
show_id*	1	8807	4404.00	2542.51	4404.0	4404.00	3264.69	1	8807	8806	0.00	-1.20	27.09
type*	2	8807	1.30	0.46	1.0	1.25	0.00	1	2	1	0.85	-1.27	0.00
title*	3	8807	4404.00	2542.51	4404.0	4404.00	3264.69	1	8807	8806	0.00	-1.20	27.09
director*	4	8807	2915.70	1426.89	3317.0	3057.86	1481.12	1	4529	4528	-0.54	-1.13	15.20
cast*	5	8807	4166.60	2337.44	4243.0	4232.99	3156.46	1	7693	7692	-0.12	-1.30	24.91
country*	6	8807	457.07	205.06	507.0	470.27	170.50	1	749	748	-0.52	-0.89	2.19
date_added*	7	8797	899.55	497.26	914.0	901.28	649.38	1	1767	1766	-0.03	-1.20	5.30
release_year	8	8807	2014.18	8.82	2017.0	2016.03	2.97	1925	2021	96	-3.45	16.22	0.09
rating*	9	8803	11.01	1.96	12.0	11.09	2.97	1	17	16	-0.42	0.38	0.02
duration*	10	8804	94.75	88.18	55.5	91.18	80.80	1	220	219	0.25	-1.69	0.94
listed_in*	11	8807	273.40	131.06	290.0	278.41	131.95	1	514	513	-0.31	-0.76	1.40
description*	12	8807	4386.75	2532.81	4386.0	4386.36	3249.86	1	8775	8774	0.00	-1.20	26.99

1.3 Exploratory Data Analysis

1.3.1 Movie vs. TV show

Table 3: Summary of Content Types on Netflix

type	n	percent
Movie	6131	0.7
TV Show	2676	0.3

Content Trends by Bar graph

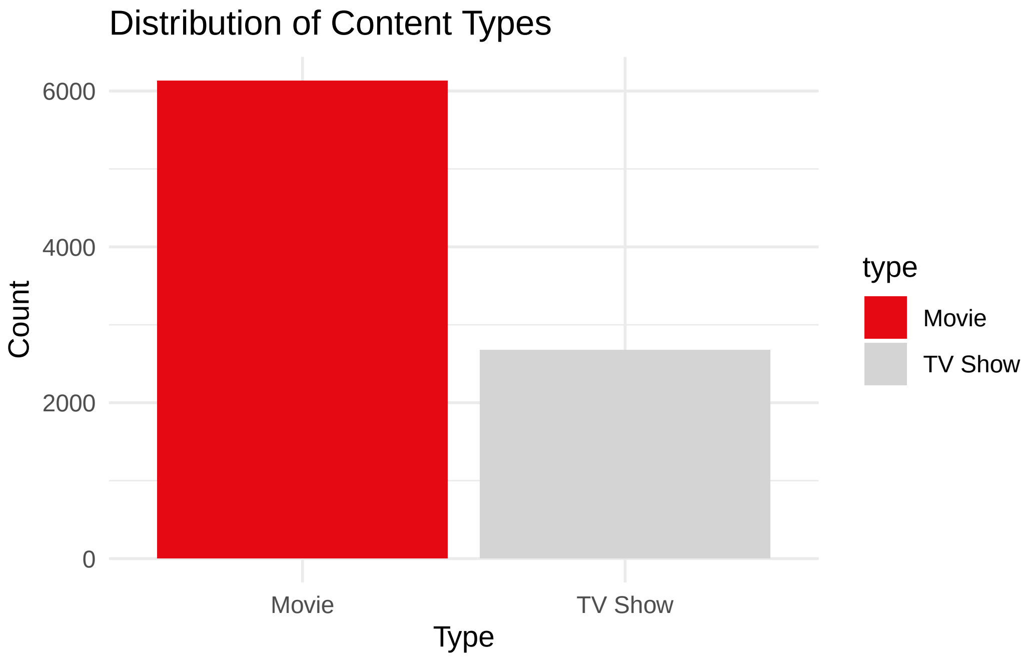


Figure 1: Content Trends by Bar graph

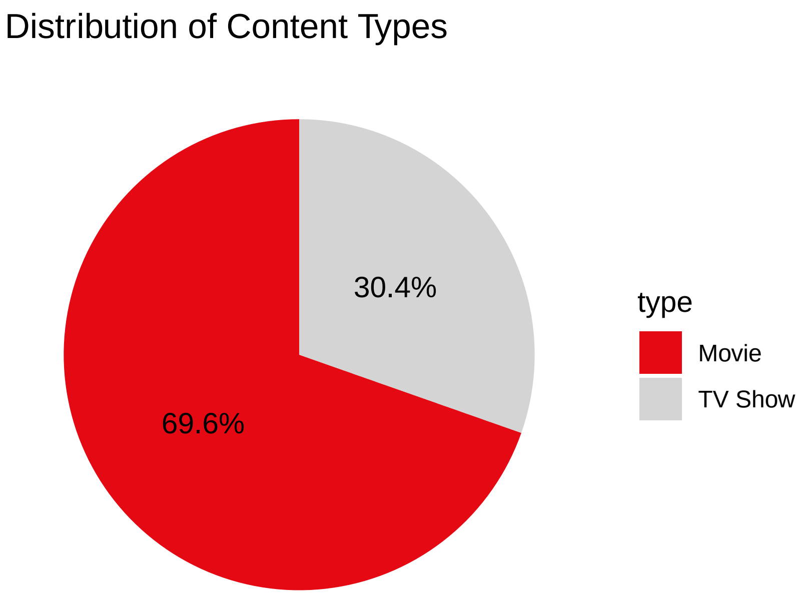


Figure 2: Content Trends by Pie Chart

Table 4: Descriptive Statistics by Content Type

type	avg_duration	median_duration	min_duration	max_duration	count
Movie	99.577187	98	3	312	6131
TV Show	1.764948	1	1	17	2676

- Those visualizations provide insights into the distribution of content types on Netflix. The bar graph shows that the majority of Netflix's content consists of Movies, which account for over 6,000 entries, while TV Shows represent a smaller portion with around 2,600 entries. The pie chart reinforces this trend, showing that Movies constitute 69.6% of the total content, while TV Shows make up 30.4%.
- In addition, the table further breaks down the descriptive statistics for each content type. Movies have an average duration of approximately 99.6 minutes with a maximum duration of 312 minutes, while TV Shows average about 1.76 seasons, with the maximum number of seasons reaching 17.

1.3.2 Movie Durations

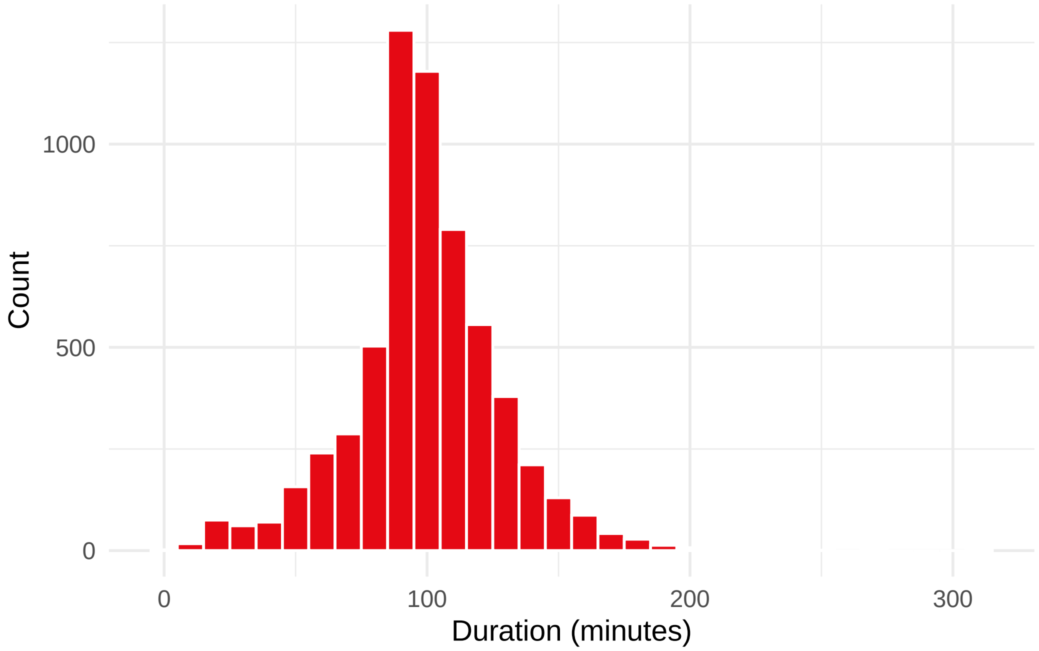


Figure 3: Distribution of movie durations in Netflix's catalog.

Table 5: Descriptive Statistics for Movie Duration

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	8804	69.85	50.81	88	69.03	44.48	1	312	311	-0.19	-1.08	0.54

Distribution of movie durations in Netflix's catalog.

- This result highlights the most common movie lengths, with the majority clustering around 100 minutes.

1.3.3 Yearly Content Addition

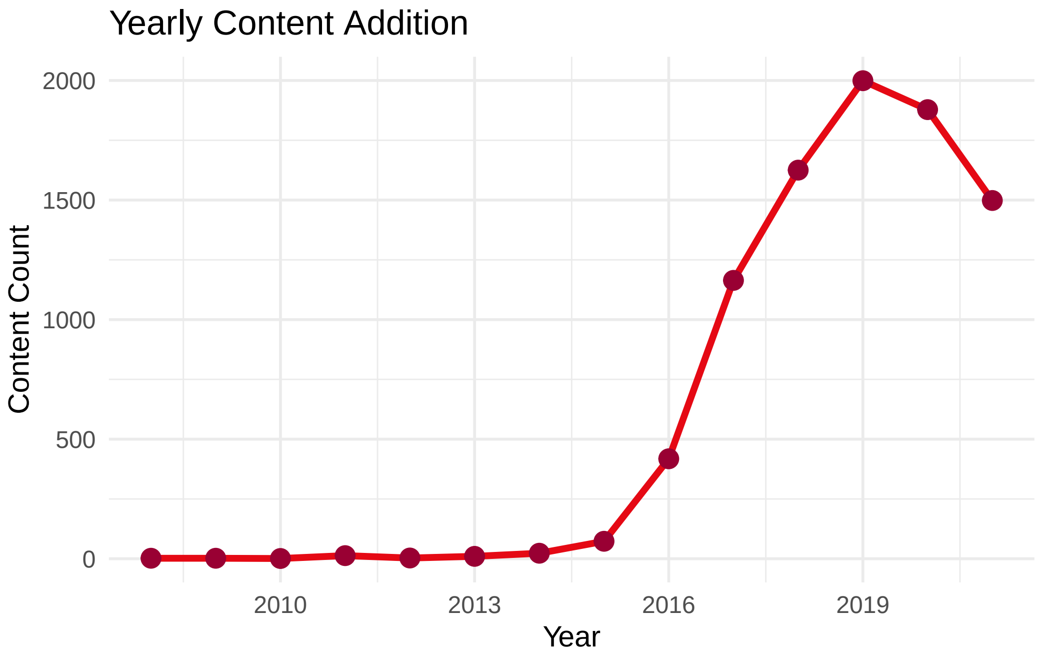


Figure 4: Yearly content addition on Netflix over time.

- This graph shows that starting in 2016, there was a sharp increase, reaching a peak in 2019 with over 2000 new entries.

1.3.4 Top 10 Genres

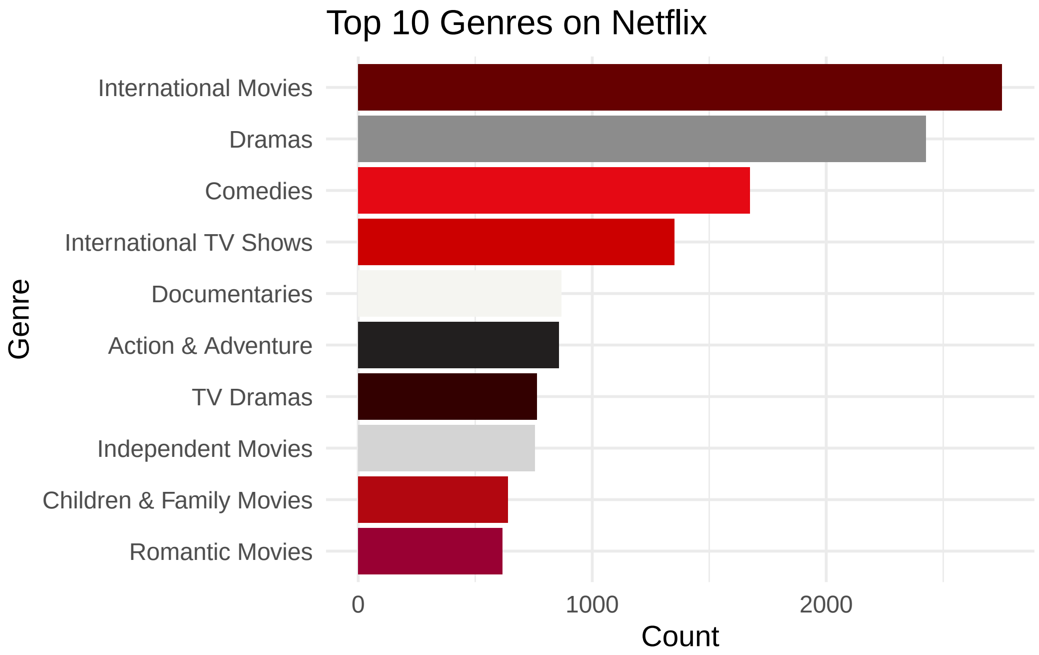


Figure 5: Genre Trends

- This bar chart indicates the most frequent genres of content available on Netflix. International Movies lead the list, followed closely by Dramas and Comedies, highlighting Netflix's global focus and strong appeal to audiences who favor story-driven entertainment. Other popular genres include International TV Shows, Documentaries, and Action & Adventure, showing diversity in Netflix's catalog.

1.3.5 Top 10 Countries

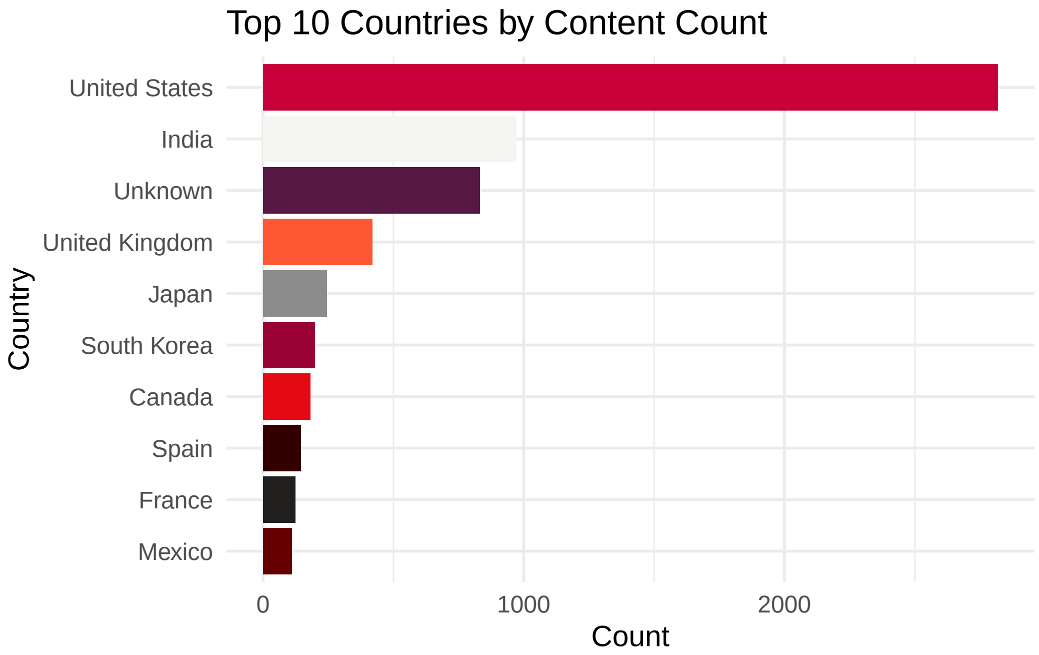


Figure 6: Top 10 Countries by Content Count

- According to this graph, the United States dominates with the highest content count by a large margin, followed by India.

1.3.6 Top 10 Actors who have appeared in the most Netflix

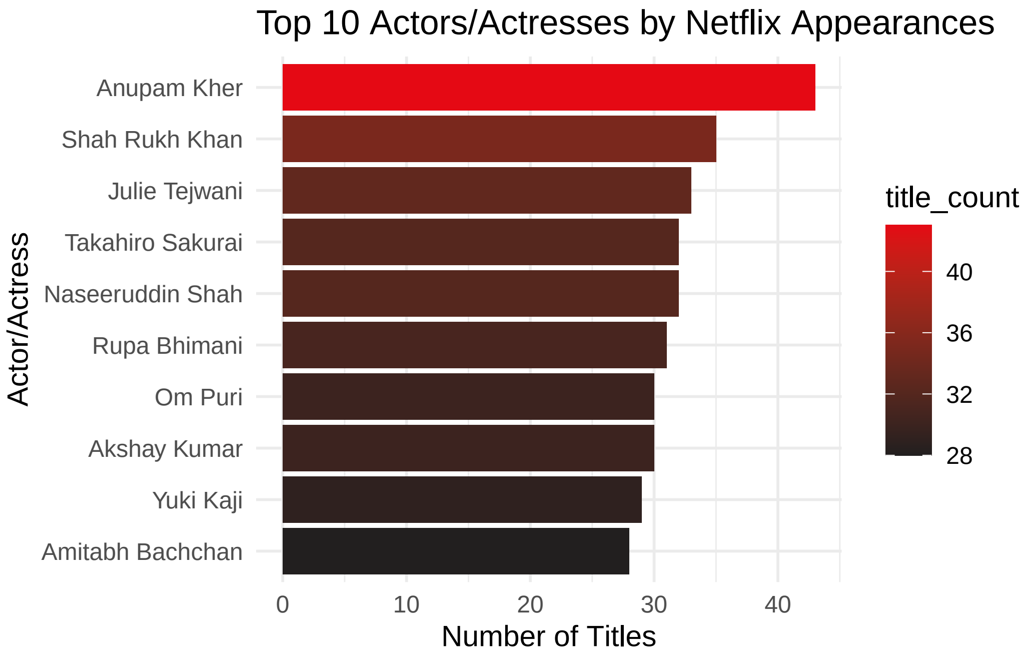


Figure 7: Top 10 Actors

- This graph highlights that the top 10 actors who frequently appear in Netflix titles.

2 Research Questions

2.1 RQ1. What are the most frequently occurring keywords in the 'description' text of Netflix content?

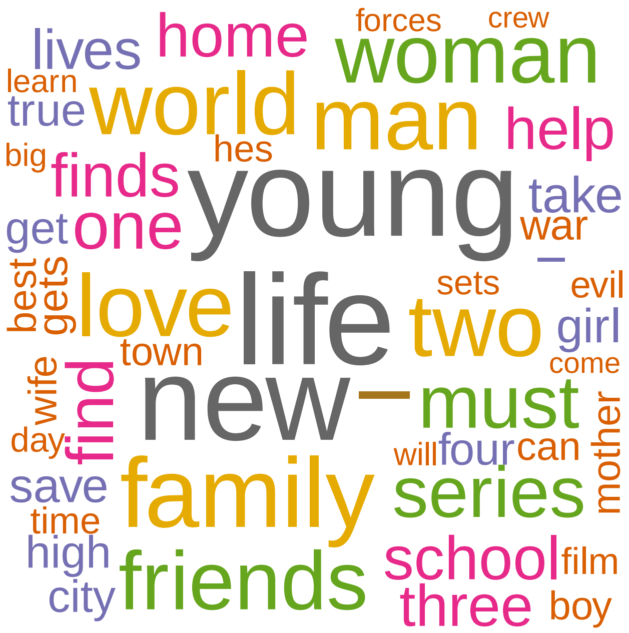


Figure 8: Most frequent words in description

2.1.1 RQ1 Findings

By analyzing frequency of text data efficiently, we used 'word cloud' graph. From research question 1, we found that the word "life" appears prominently, suggesting that many descriptions revolve around stories about human experiences. Similarly, the frequent occurrence of "young", "new", "love" and "family" may indicate Netflix's emphasis on content for family-friendly audiences or younger demographics.

2.2 RQ2. Which genres are the most frequently produced on Netflix?

Table 6: Top 10 Most Frequent Genres on Netflix

Genre	Count
International Movies	2752

Dramas	2427
Comedies	1674
International TV Shows	1351
Documentaries	869
Action & Adventure	859
TV Dramas	763
Independent Movies	756
Children & Family Movies	641
Romantic Movies	616

Top 10 most frequently produced genres on Netflix

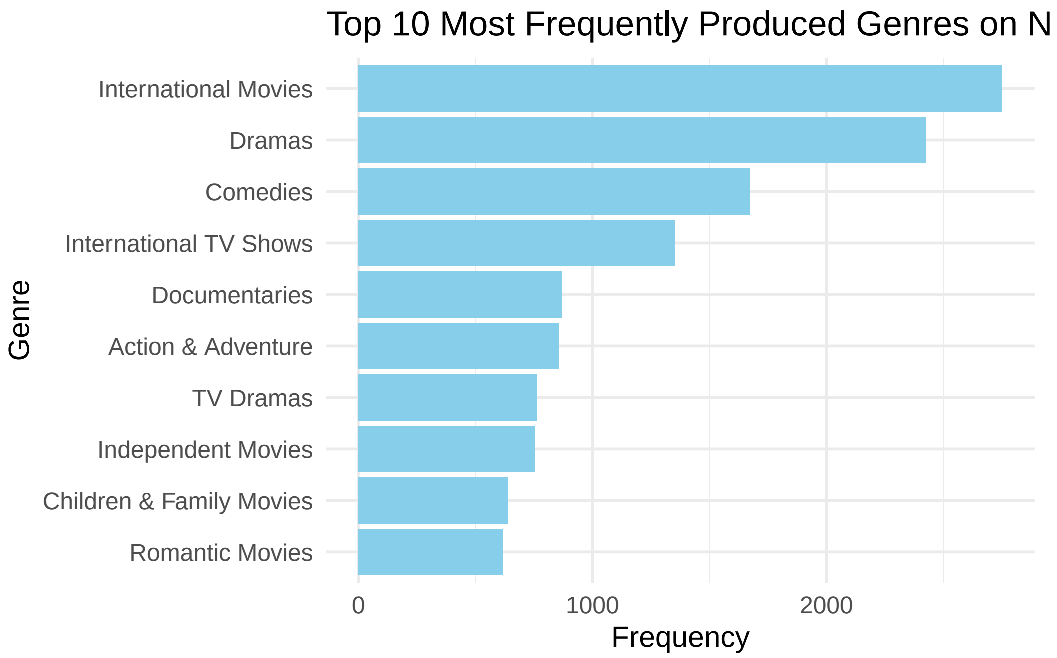


Figure 9: Top 10 most frequently produced genres on Netflix

2.2.1 RQ2 Findings

By analyzing the frequency of genres efficiently, we used a bar chart visualization. From Research Question 2, we found that the genre “Drama” appears prominently, suggesting that Netflix prioritizes producing content that often features complex narratives and emotional depth. Similarly, the frequent occurrence of genres such as “Comedy”, “Documentary”, and “Action” may indicate Netflix’s emphasis on catering to a wide range of audience preferences, from light-hearted entertainment to thought-provoking and thrilling experiences. This trend highlights Netflix’s strategy of offering diverse content to maintain its global appeal.

2.3 RQ3. Is there a correlation between specific directors and certain keywords in Netflix descriptions

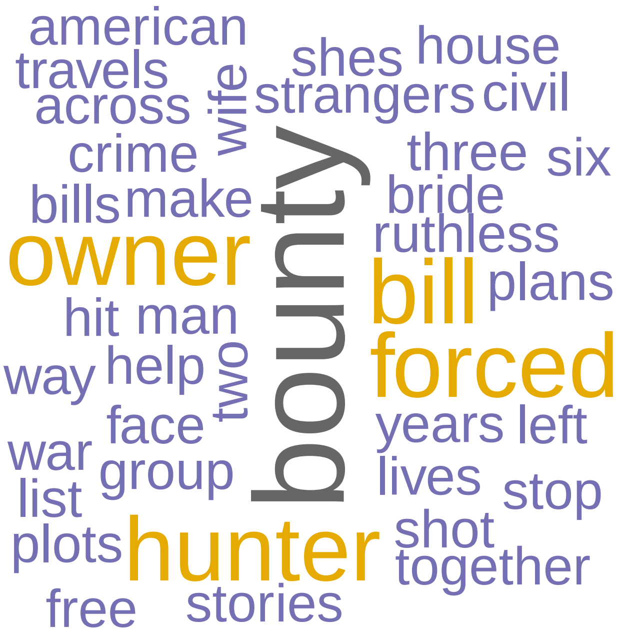


Figure 10: Corrlation between directors and keywords

2.3.1 RQ3 Findings

By analyizing the results of the data, we can concur that certain directors do end up being associated with specific directors. A notable example is the director Quentin Tarantino and the tendency for the descriptions of his movies to include words such as “violence” or “revenge”. Coincidentally, the type of movies he generally produces are actions films. From this we can gleam that certain directors will have a preference for certain genres of films.

3 Discussion

The length of movies seems to be an hour and 30 minutes long at average because it is neither too long nor too short for the average person to binge watch on any given day. Most media on Netflix is movies because that seems to be the most easily bingeable type of media. Combined with the hour and 30 minute length, this means that Netflix’s general plan is to get viewers to binge as many relatively short movies as possible. Films have historically also been known to cost less to produce than shows, as they take up less time to produce and are as a result much easier to produce more of than multi-episode shows. A large amount of media on Netflix are family friendly films because it is the easiest genre to market to the most amount of people and can provide much broader appeal with its diverse list of easy to understand and emotional themes.

4 Reference

4.1 Bansal, S. (2023). *Netflix Movies and TV Shows Dataset*. https://www.kaggle.com/datasets/shivamb/netflix-shows

5 Code Appendix

```
# Necessary libraries for the entire code
library(dplyr)
library(tidyr)
library(ggplot2)
library(psych)
library(janitor)
library(knitr)
library(kableExtra)
library(tm)
library(wordcloud)
# Load dataset
file_path <- "netflix_titles.csv"
netflix_data <- read.csv(file_path)
# summary(netflix_data)

# Use psych::describe() for summary
describe_summary <- psych::describe(netflix_data)

# Display the table
kable(describe_summary,
      caption = "Descriptive Statistics for Entire Dataset",
      digits = 2,
      format = "latex",
      booktabs = TRUE) %>%
  kable_styling(latex_options = c("hold_position", "scale_down"),
                font_size = 8) %>%
  column_spec(1, width = "2cm") %>%
  column_spec(2, width = "1.5cm") %>%
  column_spec(3:12, width = "1.2cm")

# Find Missing Data
netflix_data[netflix_data == ""] <- NA
netflix_data[netflix_data == " "] <- NA
#colSums(is.na(netflix_data))

# Calculate the number of missing values in each column
missing_data <- colSums(is.na(netflix_data))
# print(missing_data)
missing_data_df <- data.frame(Column = names(missing_data), MissingCount = missing_data)

ggplot(missing_data_df, aes(x = reorder(Column, -MissingCount), y = MissingCount, fill = M
  geom_bar(stat = "identity") +
  coord_flip() +
  theme_minimal() +
  labs(
    title = "Missing Data Count by Column",
    x = "Columns",
    y = "Count of Missing Values"
  ) +
  scale_fill_gradient(low = "blue", high = "red") +
  theme(legend.position = "none")
# Cleaning Missing Data
  # Replace missing values with "Unknown"
netflix_data$director[is.na(netflix_data$director)] <- "Unknown"
netflix_data$cast[is.na(netflix_data$cast)] <- "Unknown"
netflix_data$country[is.na(netflix_data$country)] <- "Unknown"
# return(netflix_data)
#colSums(is.na(netflix_data))

# Save the cleaned dataset
write.csv(netflix_data, "cleaned_netflix_data.csv", row.names = FALSE)
cleaned_data <- read.csv("cleaned_netflix_data.csv")
# Descriptive statistics for the cleaned dataset
describe_stats <- psych::describe(cleaned_data)

# Display the results using kable
kable(describe_stats,
      caption = "Descriptive Statistics for Cleaned Dataset",
      digits = 2,
      format = "latex",
      booktabs = TRUE) %>%
  kable_styling(latex_options = c("hold_position", "scale_down"),
                font_size = 8) %>%
  column_spec(1, width = "2cm") %>%
  column_spec(2, width = "1.5cm") %>%
  column_spec(3:12, width = "1.2cm")


# Tabyl for summary of 'type' column
type_summary <- netflix_data %>%
  tabyl(type)

# Display the result
kable(type_summary, caption = "Summary of Content Types on Netflix", digits = 2)


# 1. Bar graph
ggplot(cleaned_data, aes(x = type, fill = type)) +
  geom_bar() +
  scale_fill_manual(values = c("#e50914", "#d4d4d4")) +
  labs(title = "Distribution of Content Types", x = "Type", y = "Count") +
  theme_minimal()
# 2. Pie graph
# Calculate percentages
content_type_dist <- cleaned_data %>%
  count(type) %>%
  mutate(percentage = n / sum(n) * 100)
# Visualization
ggplot(content_type_dist, aes(x = "", y = n, fill = type)) +
  geom_bar(width = 1, stat = "identity") +  # Bar chart as base
  coord_polar(theta = "y") +                # Convert to pie chart
  scale_fill_manual(values = c("#e50914", "#d4d4d4")) +
  geom_text(aes(label = paste0(round(percentage, 1), "%")),  # Add percentage labels
            position = position_stack(vjust = 0.5)) +     # Center labels
  labs(title = "Distribution of Content Types", x = NULL, y = NULL) +
  theme_minimal() +
  theme(axis.text.x = element_blank(),          # Remove x-axis labels
        axis.ticks = element_blank(),
        panel.grid = element_blank())           # Remove grid lines
# Create the duration col
```

```r
cleaned_data <- cleaned_data %>%
  mutate(duration_numeric = as.numeric(gsub("[^0-9]", "", duration)))

grouped_stats <- cleaned_data %>%
  group_by(type) %>%
  summarize(
    avg_duration = mean(duration_numeric, na.rm = TRUE),
    median_duration = median(duration_numeric, na.rm = TRUE),
    min_duration = min(duration_numeric, na.rm = TRUE),
    max_duration = max(duration_numeric, na.rm = TRUE),
    count = n()
  )

# Display the table using knitr::kable
kable(grouped_stats, caption = "Descriptive Statistics by Content Type")


# Extract numeric values from duration
cleaned_data$duration_numeric <- as.numeric(gsub("[^0-9]", "", cleaned_data$duration))

# Distribution of duration for movies
movie_duration <- cleaned_data %>%
  filter(type == "Movie")

ggplot(movie_duration, aes(x = duration_numeric)) +
  geom_histogram(binwidth = 10, fill = "#e50914", color = "white") +
  labs( x = "Duration (minutes)", y = "Count") +
  theme_minimal()

# Get descriptive statistics for numeric columns
describe_stats <- psych::describe(cleaned_data$duration_numeric)

# Display the results with kable
kable(describe_stats,
      caption = "Descriptive Statistics for Movie Duration",
      digits = 2,
      align = "c")

# Yearly trend
cleaned_data$year_added <- as.numeric(format(as.Date(cleaned_data$date_added, "%B %d, %Y")
# Year-wise content count
yearly_content <- cleaned_data %>%
  filter(!is.na(year_added)) %>%
  count(year_added)

# Line plot
ggplot(yearly_content, aes(x = year_added, y = n)) +
  geom_line(color = "#e50914", size = 1.2) +
  geom_point(color = "#990033", size = 3) +
  labs(title = "Yearly Content Addition", x = "Year", y = "Content Count") +
  theme_minimal()

genre_data <- cleaned_data %>%
  separate_rows(listed_in, sep = ", ") %>%
  count(listed_in, sort = TRUE) %>%
  slice_max(order_by = n, n = 10)

# Plot top genres
ggplot(genre_data, aes(x = reorder(listed_in, n), y = n, fill = listed_in)) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values = c("#221f1f", "#b20710", "#e50914", "#f5f5f1",
                               "#8c8c8c", "#d4d4d4", "#660000", "#cc0000",
                               "#990033", "#330000")) +
  coord_flip() +
  labs(title = "Top 10 Genres on Netflix", x = "Genre", y = "Count") +
  theme_minimal() +
  theme(legend.position = "none")
# Find the top 15 countries
top_countries <- cleaned_data %>%
  count(country, sort = TRUE) %>%
  top_n(10, n)
  # Visualization
custom_colors <- c("#e50914", "#221f1f", "#f5f5f1", "#8c8c8c",
                   "#660000", "#990033", "#330000",
                   "#ff5733", "#c70039", "#581845")

ggplot(top_countries, aes(x = reorder(country, n), y = n, fill = country)) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values = custom_colors) +  # Apply custom colors
  coord_flip() +
  labs(title = "Top 10 Countries by Content Count", x = "Country", y = "Count") +
  theme_minimal()+
  theme(legend.position = "none")
# Split the 'cast' column and count occurrences of actors/actresses
top_actors <- cleaned_data %>%
  filter(!is.na(cast) & cast != "Unknown") %>%
  separate_rows(cast, sep = ",\\s*") %>%
  group_by(cast) %>%
  summarize(title_count = n()) %>%
  arrange(desc(title_count)) %>%
  slice_head(n = 10)  # Top 10 actors/actresses

# Visualization: Bar plot of top actors/actresses
ggplot(top_actors, aes(x = reorder(cast, title_count), y = title_count, fill = title_count
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 Actors/Actresses by Netflix Appearances",
       x = "Actor/Actress", y = "Number of Titles") +
  scale_fill_gradient(low = "#221f1f", high = "#e50914") +
  theme_minimal()
# Extract the 'description' column and remove missing values
descriptions <- cleaned_data$description
descriptions <- na.omit(descriptions)

# Combine all descriptions into a single text
text <- paste(descriptions, collapse = " ")

# Create a corpus
corpus <- Corpus(VectorSource(text))

# Preprocess the text: remove punctuation, numbers, stopwords, and convert to lowercase
corpus <- corpus %>%
  tm_map(content_transformer(tolower)) %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(removeWords, stopwords("english"))

# Create a term-document matrix
tdm <- TermDocumentMatrix(corpus)
tdm_matrix <- as.matrix(tdm)

# Sum the frequency of words
```

```
word_freq <- sort(rowSums(tdm_matrix), decreasing = TRUE)

# Create a data frame of words and their frequencies
word_data <- data.frame(word = names(word_freq), freq = word_freq)

# Generate the word cloud
set.seed(123) # For consistent result
wordcloud(
  words = word_data$word,
  freq = word_data$freq,
  min.freq = 3, # Set minimum frequency for words to appear
  max.words = 150, # Set the maximum number of words to display
  random.order = FALSE,
  colors = brewer.pal(8, "Dark2")
)

# Step 1: Extract and preprocess data
# Load the 'listed_in' column, which contains genre information, and remove missing values
genres <- cleaned_data$listed_in
genres <- na.omit(genres)

# Step 2: Split multiple genres in a single entry into separate rows
# Split genres by comma and trim whitespace
genres_data <- data.frame(genres = unlist(strsplit(as.character(genres), ",\\s*")))

# Step 3: Count the frequency of each genre
genre_frequency <- genres_data %>%
  group_by(genres) %>%
  summarise(Frequency = n()) %>%
  arrange(desc(Frequency))

# Display the top 10 genres
# print(head(genre_frequency, 10))
kable(head(genre_frequency, 10),
      caption = "Top 10 Most Frequent Genres on Netflix",
      col.names = c("Genre", "Count"),
      digits = 0,
      format = "markdown") %>%
  kable_styling(latex_options = c("hold_position", "striped"), font_size = 8)

# Step 4: Visualize the top genres
```

```
ggplot(data = head(genre_frequency, 10), aes(x = reorder(genres, Frequency), y = Frequency
  geom_bar(stat = "identity", fill = "skyblue") +
  coord_flip() +
  labs(title = "Top 10 Most Frequently Produced Genres on Netflix",
       x = "Genre",
       y = "Frequency") +
  theme_minimal()

# Step 5: Save the result for further use
write.csv(genre_frequency, "netflix_genre_frequency.csv", row.names = FALSE)

# Additional Exploration: Genres across regions
# If there are columns related to country or region, we can analyze genre distributions by
if ("country" %in% colnames(cleaned_data)) {
  genre_region <- cleaned_data %>%
    select(listed_in, country) %>%
    separate_rows(listed_in, sep = ",\\s*") %>%
    group_by(country, listed_in) %>%
    summarise(Frequency = n()) %>%
    arrange(country, desc(Frequency))

  # Save region-specific genre data
  write.csv(genre_region, "netflix_genre_by_region.csv", row.names = FALSE)
}


# Preprocessing to find correlation between directors and keywords in descriptions

# Step 1: Extract and preprocess data
cleaned_data$description <- tolower(cleaned_data$description)
cleaned_data$description <- gsub("[[:punct:]]", "", cleaned_data$description)
cleaned_data$description <- gsub("[[:digit:]]", "", cleaned_data$description)
cleaned_data$description <- gsub("\\s+", " ", cleaned_data$description)
cleaned_data$description <- trimws(cleaned_data$description)

# Remove stopwords from descriptions
stop_words <- stopwords("en")
cleaned_data$processed_description <- sapply(cleaned_data$description, function(x) {
  paste(setdiff(unlist(strsplit(x, " ")), stop_words), collapse = " ")
})
```

```r
# Step 2: Create a mapping of directors to their descriptions
director_keywords <- cleaned_data %>%
  filter(director != "Unknown") %>%
  group_by(director) %>%
  summarise(all_descriptions = paste(processed_description, collapse = " ")) %>%
  ungroup()

# Step 3: Tokenize and extract keywords for each director
corpus <- Corpus(VectorSource(director_keywords$all_descriptions))

# Further preprocessing
corpus <- corpus %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(removeWords, stopwords("english"))

# Create a Term-Document Matrix
tdm <- TermDocumentMatrix(corpus)
tdm_matrix <- as.matrix(tdm)

# Find the most frequent terms for each director
director_keywords$top_keywords <- apply(tdm_matrix, 2, function(x) {
  top_terms <- names(sort(x, decreasing = TRUE)[1:10]) # Top 10 terms
  paste(top_terms, collapse = ", ")
})

# Step 4: Visualize the correlations
selected_director <- "Quentin Tarantino" # Replace with any director of interest

# Find the corresponding keywords
if (selected_director %in% director_keywords$director) {
  selected_keywords <- tdm_matrix[, which(director_keywords$director == selected_director)
  selected_keywords <- selected_keywords[selected_keywords > 0]

  # Create a word cloud
  wordcloud(
    words = names(selected_keywords),
    freq = selected_keywords,
    min.freq = 1,
    max.words = 100,
    random.order = FALSE,
    colors = brewer.pal(8, "Dark2")
  )
} else {
  cat("Director not found in the dataset.")
}

# Step 5: Save results
write.csv(director_keywords, "director_keywords.csv", row.names = FALSE)
```