

# Usenet Project - CSV Cleaning

Emerson Johnston

## Maintenance

```
rm(list = ls())
knitr::opts_knit$set(root.dir = '/Users/emerson/Github/usenet_webpage')

# Load Libraries
library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.5.1      v tibble    3.2.1
## v lubridate  1.9.3      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

library(ggplot2)
library(dplyr)
library(readr)
library(syuzhet)

# Directories
output_directory <- "/Users/emerson/Github/usenet_webpage"
threads_directory <- file.path(output_directory, "CSV Files/Threads")
comments_directory <- file.path(output_directory, "CSV Files/Comments")

# Load the datasets
all_threads <- read.csv(file.path(threads_directory, "combined_threads.csv"))
all_comments <- read.csv(file.path(comments_directory, "combined_comments_AS.csv"))
```

## Dataset 1 - All Comments Cleaned

```
# Map newsgroups to IDs
newsgroup_ids <- c("netmed" = "NG01", "netmotss" = "NG02", "netnews" = "NG03",
                  "netpolitics" = "NG04", "netreligion" = "NG05", "netsingles" = "NG06")
```

```

# Threads cleaning
all_threads <- all_threads %>%
  mutate(NG_ID = factor(newsgroup, levels = names(newsgroup_ids), labels = newsgroup_ids),
         NG_TH_ID = paste(NG_ID, ThreadID, sep = "_")) %>%
  rename(TH_ID = ThreadID) %>%
  select(NG_TH_ID, TH_ID, NG_ID, everything()) %>%
  mutate(Date = as.Date(Date, format = "%m/%d/%y"))

# Comments cleaning
all_comments <- all_comments %>%
  rename(
    TH_CM_ID = Unique.Comment.ID,
    TH_ID = Thread.ID,
    CM_ID = Comment.ID) %>%
  mutate(NG_ID = factor(newsgroup, levels = names(newsgroup_ids), labels = newsgroup_ids),
         NG_TH_CM_ID = paste(NG_ID, TH_CM_ID, sep = "_"),
         NG_TH_ID = paste(NG_ID, TH_ID, sep = "_")) %>%
  select(NG_TH_CM_ID, NG_TH_ID, TH_CM_ID, CM_ID, TH_ID, NG_ID, everything()) %>%
  mutate(Date.and.Time = as.POSIXct(gsub("[^[:alnum:]][:punct:]", "", Date.and.Time), format = "%b %d,
    Hour = as.numeric(format(Date.and.Time, "%H")),
    Date = as.Date(Date.and.Time))

# Add sentiment scores
all_comments <- all_comments %>%
  mutate(SentimentScore = get_sentiment(Full.Text, method = "afinn"))

all_comments <- all_comments %>%
  mutate(
    Author = case_when(
      Author == "Cowenton Volunteer Fire Department" ~ "Ron Natalie",
      TRUE ~ Author
    )
  )

write.csv(all_threads, file.path(threads_directory, "dataset1_threads.csv"), row.names = FALSE)
write.csv(all_comments, file.path(comments_directory, "dataset1_comments.csv"), row.names = FALSE)

```

## Dataset 2 - AIDS-Related Threads and their Associated Comments, as Determined by Titles (1982–1986)

```

# Define AIDS-related keywords
aids_keywords <- c("aids", "acquired immune deficiency syndrome", "human immunodeficiency virus",
  "gay-related immune deficiency", "gay plague",
  "hiv", "htlv", "human t-lymphotropic virus", "gay cancer", "kaposi's sarcoma",
  "slim disease", "pneumocystis pneumonia", "gay disease", "homosexual disease")

# Step 1: Filter threads by `Thread.Title` containing AIDS-related keywords
relevant_threads <- all_threads %>%
  filter(str_detect(tolower(Thread.Title), paste(tolower(aids_keywords), collapse = "|")))

# Step 2: Get `NG_TH_ID`s for relevant threads

```

```

relevant_thread_ids <- relevant_threads$NG_TH_ID

# Step 3: Filter comments associated with these threads
relevant_comments <- all_comments %>%
  filter(NG_TH_ID %in% relevant_thread_ids)

write.csv(relevant_threads, file.path(threads_directory, "dataset2_threads.csv"), row.names = FALSE)
write.csv(relevant_comments, file.path(comments_directory, "dataset2_comments.csv"), row.names = FALSE)

```

### Dataset 3 - AIDS-Related Threads and their Associated Comments (1982–1986) - Filtered By Influential Authors Comments

```

# Load required libraries
library(dplyr)
library(igraph)

##
## Attaching package: 'igraph'

## The following objects are masked from 'package:lubridate':
##
##    %--%, union

## The following objects are masked from 'package:dplyr':
##
##    as_data_frame, groups, union

## The following objects are masked from 'package:purrr':
##
##    compose, simplify

## The following object is masked from 'package:tidyr':
##
##    crossing

## The following object is masked from 'package:tibble':
##
##    as_data_frame

## The following objects are masked from 'package:stats':
##
##    decompose, spectrum

## The following object is masked from 'package:base':
##
##    union

```

```

# Step 2: Create an author co-participation network
author_pairs <- relevant_comments %>%
  filter(!is.na(TH_ID)) %>%
  group_by(TH_ID) %>%
  summarise(
    Pairs = list(if (length(unique(Author)) > 1) {
      as.data.frame(t(combn(unique(Author), 2)))
    } else {
      NULL
    })
  ) %>%
  unnest(Pairs, keep_empty = TRUE) %>%
  rename(Author1 = V1, Author2 = V2) %>%
  count(Author1, Author2, name = "Weight") %>%
  filter(!is.na(Author1) & !is.na(Author2))

# Step 3: Create a graph from the author pairs
author_network <- graph_from_data_frame(author_pairs, directed = FALSE)

# Step 4: Identify influential authors using degree centrality
degree_centrality <- strength(author_network, mode = "all", weights = E(author_network)$Weight)

# Create a data frame of authors and their influence scores
influential_authors <- data.frame(
  Author = names(degree_centrality),
  InfluenceScore = degree_centrality
) %>%
  arrange(desc(InfluenceScore)) %>%
  head(20) # Select the top 20 influential authors

# Save the list of influential authors
write.csv(influential_authors, file.path(output_directory, "CSV Files", "influential_authors.csv"), row.names = FALSE)

# Print the top influential authors
print(influential_authors)

```

##	Author	InfluenceScore
##	Craig Werner	108
##	Ron Rizzo	92
##	Steve Dyer	74
##	Rob Bernardo	41
##	Bill Stoll	36
##	Alan J Rosenthal	31
##	Andrew Klossner	31
##	Brian Mavrogeorge	31
##	David Sher	31
##	Harold Ancell	31
##	John Gurian	31
##	Mike Leibensperger	31
##	Pam Pincha	31
##	Rod Williams	31
##	Ron Natalie	31
##	Roy Smith	31

## Stephanie Da Silva	Stephanie Da Silva	31
## James R. Carbin	James R. Carbin	26
## Beth Christy	Beth Christy	20
## Bob Bickford	Bob Bickford	19

```

# Step 5: Filter comments authored by influential authors
influential_author_comments <- relevant_comments %>%
  filter(Author %in% influential_authors$Author)

# Step 6: Identify threads with at least one influential author
influential_threads <- relevant_threads %>%
  filter(NG_TH_ID %in% influential_author_comments$NG_TH_ID)

# Step 7: Include all comments in threads with influential authors
all_comments_in_influential_threads <- relevant_comments %>%
  filter(NG_TH_ID %in% influential_threads$NG_TH_ID)

# Step 8: Save the final datasets
write.csv(influential_threads,
  file.path(threads_directory, "dataset3_threads.csv"),
  row.names = FALSE)
write.csv(all_comments_in_influential_threads,
  file.path(comments_directory, "dataset3_comments_all.csv"),
  row.names = FALSE)
write.csv(influential_author_comments,
  file.path(comments_directory, "dataset3_comments_onlyinfluential.csv"),
  row.names = FALSE)

# Step 9: Print summary statistics
cat("Number of threads involving influential authors:", nrow(influential_threads), "\n")

```

```
## Number of threads involving influential authors: 74
```

```
cat("Number of comments in these threads (all comments):", nrow(all_comments_in_influential_threads), "\n")
```

```
## Number of comments in these threads (all comments): 266
```

```
cat("Number of comments by influential authors only:", nrow(influential_author_comments), "\n")
```

```
## Number of comments by influential authors only: 158
```

## Descriptive Statistics Tables

### Dataset 1 Descriptive Statistics

```

library(sjPlot)

# Dataset 1: All Comments
dataset1_summary <- all_comments %>%

```

```

group_by(newsgroup) %>%
  summarize(
    Threads = n_distinct(TH_ID),
    Comments = n(),
    Authors = n_distinct(Author),
    Avg_Comments_Per_Thread = Comments / Threads,
    Avg_Sentiment_Score = mean(SentimentScore, na.rm = TRUE)
  )

# Add a totals row
dataset1_totals <- dataset1_summary %>%
  summarize(
    newsgroup = "Total",
    Threads = sum(Threads),
    Comments = sum(Comments),
    Authors = n_distinct(all_comments$Author),
    Avg_Comments_Per_Thread = sum(Comments) / sum(Threads),
    Avg_Sentiment_Score = mean(all_comments$SentimentScore, na.rm = TRUE)
  )

# Combine the summary with the totals row
dataset1_summary <- bind_rows(dataset1_summary, dataset1_totals)

# Save or print the summary
tab_df(dataset1_summary, file = file.path(output_directory, "Images and Tables/Tables/dataset1_statisti

```

newsgroup

Threads

Comments

Authors

Avg\_Comments\_Per\_Thread

Avg\_Sentiment\_Score

netmed

1442

3635

1332

2.52

-0.46

netmotss

1074

2532

679

2.36

0.51

netnews

2430  
5297  
1684  
2.18  
1.57  
netpolitics  
4126  
13659  
2371  
3.31  
-1.99  
netreligion  
3042  
8016  
1490  
2.64  
1.46  
netsingles  
3504  
10752  
2626  
3.07  
3.14  
Total  
15618  
43891  
6584  
2.81  
0.60

## Dataset 2 Descriptive Statistics

```
dataset2_stats <- relevant_comments %>%  
  group_by(newsgroup) %>%  
  summarize(  
    Threads = n_distinct(TH_ID),  
    Comments = n(),  
    Authors = n_distinct(Author),
```

```

    Avg_Comments_Per_Thread = Comments / Threads,
    Avg_Sentiment_Score = mean(SentimentScore, na.rm = TRUE)
  )

# Add a totals row
dataset2_totals <- dataset2_stats %>%
  summarize(
    newsgroup = "Total",
    Threads = sum(Threads),
    Comments = sum(Comments),
    Authors = n_distinct(relevant_comments$Author),
    Avg_Comments_Per_Thread = sum(Comments) / sum(Threads),
    Avg_Sentiment_Score = mean(relevant_comments$SentimentScore, na.rm = TRUE)
  )

dataset2_stats <- bind_rows(dataset2_stats, dataset2_totals)

# Save as HTML or CSV
tab_df(dataset2_stats, file = file.path(output_directory, "Images and Tables/Tables/dataset2_statistics

```

newsgroup

Threads

Comments

Authors

Avg\_Comments\_Per\_Thread

Avg\_Sentiment\_Score

netmed

47

143

80

3.04

-3.49

netmotss

60

157

86

2.62

-3.80

netnews

8

11

9

1.38



2.45  
 netreligion  
 3  
 5  
 4  
 1.67  
 -1.80  
 netsingles  
 10  
 32  
 23  
 3.20  
 -3.50  
 Total  
 128  
 348  
 135  
 2.72  
 -3.42

### Dataset 3 Descriptive Statistics

```
library(dplyr)
library(sjPlot)

# Compute statistics grouped by newsgroup
descriptive_stats <- all_comments_in_influential_threads %>%
  group_by(newsgroup) %>%
  summarise(
    Threads = n_distinct(NG_TH_ID), # Total threads
    Total_Comments = n(), # Total comments
    Influential_Authors_Comments = sum(Author %in% influential_author_comments$Author), # Influential
    Total_Authors = n_distinct(Author), # Total unique authors
    Influential_Authors = n_distinct(Author[Author %in% influential_author_comments$Author]), # Unique
    Average_Comments_Per_Thread = n() / n_distinct(NG_TH_ID), # Avg comments per thread
    Avg_Total_Comments_Sentiment_Score = mean(SentimentScore, na.rm = TRUE), # Avg sentiment for all c
    Avg_Influential_Comments_Sentiment_Score = mean(SentimentScore[Author %in% influential_author_comme
  )

# Add a totals row
totals_row <- descriptive_stats %>%
  summarise(
    newsgroup = "Total",
```

```

Threads = sum(Threads),
Total_Comments = sum(Total_Comments),
Influential_Authors_Comments = sum(Influential_Authors_Comments),
Total_Authors = n_distinct(all_comments_in_influential_threads$Author),
Influential_Authors = n_distinct(influential_author_comments$Author),
Average_Comments_Per_Thread = sum(Total_Comments) / sum(Threads),
Avg_Total_Comments_Sentiment_Score = mean(all_comments_in_influential_threads$SentimentScore, na.rm =
Avg_Influential_Comments_Sentiment_Score = mean(influential_author_comments$SentimentScore, na.rm =
)

# Combine newsgroup stats with totals row
descriptive_stats <- bind_rows(descriptive_stats, totals_row)

# Save and display the table
tab_df(descriptive_stats, file = file.path(output_directory, "Images and Tables/Tables/dataset3_statist.

```

```

newsgroup
Threads
Total_Comments
Influential_Authors_Comments
Total_Authors
Influential_Authors
Average_Comments_Per_Thread
Avg_Total_Comments_Sentiment_Score
Avg_Influential_Comments_Sentiment_Score
netmed
33
125
73
63
20
3.79
-3.89
-5.14
netmotss
37
122
76
60
20
3.30

```

-4.81  
 -4.80  
 netsingles  
 4  
 19  
 9  
 12  
 4  
 4.75  
 -3.79  
 -6.22  
 Total  
 74  
 266  
 158  
 79  
 20  
 3.59  
 -4.30  
 -5.04

## Dataset 3 Author Statistics

```
# Step 1: Join influential_authors to dataset3_influential_comments
dataset3_influential_comments <- influential_author_comments %>%
  left_join(influential_authors, by = "Author") # Join InfluenceScore based on Author

# Step 2: Add descriptive statistics for each influential author
author_stats <- dataset3_influential_comments %>%
  group_by(Author) %>%
  summarise(
    Influence = first(InfluenceScore), # Influence is now directly available
    Num_Comments = n_distinct(TH_CM_ID), # Number of comments authored
    Num_Threads = n_distinct(NG_TH_ID), # Number of threads participated in
    Threads_Started = sum(CM_ID == "CM00001", na.rm = TRUE), # Threads started
    Avg_Sentiment = mean(SentimentScore, na.rm = TRUE) # Average sentiment score
  )

# Step 3: Filter authors with missing Influence scores and sort by Influence
author_stats <- author_stats %>%
  filter(!is.na(Influence)) %>%
  arrange(desc(Influence))
```

```
# Step 4: Save the table as an HTML file
tab_df(author_stats, file = file.path(output_directory, "Images and Tables/Tables/dataset3_author_statist
```

Author
Influence
Num_Comments
Num_Threads
Threads_Started
Avg_Sentiment
Craig Werner
108
42
40
28
-1.98
Ron Rizzo
92
35
25
17
-10.40
Steve Dyer
74
16
16
3
-4.06
Rob Bernardo
41
19
14
6
-4.89
Bill Stoll
36
5
4

2  
-15.00  
Alan J Rosenthal  
31  
2  
2  
0  
-3.00  
Andrew Klossner  
31  
2  
2  
0  
-7.00  
Brian Mavrogeorge  
31  
4  
2  
0  
-4.50  
David Sher  
31  
2  
2  
0  
-9.00  
Harold Ancell  
31  
2  
2  
0  
-6.00  
John Gurian  
31  
4  
2

0  
4.00  
Mike Leibensperger  
31  
2  
2  
0  
9.00  
Pam Pincha  
31  
4  
4  
0  
-2.00  
Rod Williams  
31  
3  
3  
1  
-4.67  
Ron Natalie  
31  
2  
2  
0  
-11.00  
Roy Smith  
31  
2  
2  
0  
4.00  
Stephanie Da Silva  
31  
2  
2

0  
0.00  
James R. Carbin  
26  
4  
4  
0  
-6.00  
Beth Christy  
20  
4  
4  
2  
-5.00  
Bob Bickford  
19  
2  
2  
0  
-1.00