Project Phase 1

Baktash Ansari

June 6, 2023

1 Introduction

In this report, I intend to present the stages of data extraction and analysis, along with the reports obtained from the code.

Tasks performed

- Data crawling in multiple stages:
  - Crawling the labels
  - Crawling the IMDb ID of each movie
  - Crawling the subtitles of each movie
- Data cleaning:
  - Actions taken for data cleaning
  - Location of the cleaned data storage
- Data segmentation and dataframe creation:
  - Segmentation into sentences and words
  - Creation of a dataframe for uploading to Hugging Face
- Preparation of reports:
  - Reports in the form of tables and charts
- Final remarks and challenges

2 Crawling Data

To crawl the data, I first provide an explanation about the data structure used. Since I intend to separate the subtitles based on age rating, I need to collect a set of movies to be able to separate the age rating of each movie and its corresponding subtitle. For this purpose, I use a unique value for each movie that the IMDb website has assigned, called the IMDb ID. This value is a unique key that distinguishes each movie from another, and we need it to extract information about each movie. To crawl a large collection of IMDb IDs, I used the Beautiful Soup library, which allowed me to parse the HTML of the IMDb website and obtain approximately 5,000 to 10,000 IDs.

Next, after crawling these IDs, using them and the cinemagoer library in Python, which is an IMDb-dependent library, I was able to extract the certificates of each movie. A certificate is essentially a list of movie age ratings in different countries, according to the different laws of those countries. I used the United States as the country of reference.

In the second part, by taking the number of subtitles from the user for data crawling and IMDb IDs, I crawled a collection of each IMDb ID and its corresponding age rating, and stored these values in a text file named "labels.txt."

In the third step, it is necessary to download the subtitles for each movie based on the IMDb ID. For this task, I used the library and API associated with the Open Subtitles website. An important note about downloading subtitles with each account is that only 300 subtitles can be crawled per day with a single account. By creating two "Maximus" accounts, I can download up to 600 subtitles within 24 hours.

Finally, I downloaded the subtitles and stored them in the "subtitle/eng" folder.

3 Structure of crawled data

The structure of crawled data is as follows:

- A folder named "subtitle" where the subtitles are stored (the name of each subtitle is equal to its corresponding IMDb ID).
- A file named "labels.txt" where the IDs and their corresponding labels are stored.
- A file where the IDs of the movies for which subtitles have been downloaded are placed.

4 Cleaning Data

I have performed the following steps to clean the data:

- Since subtitle files are in the SRT format and have a specific structure where each sentence is displayed with a time stamp and sequence number, I need to remove these values and keep only the subtitle text. To accomplish this, I use the `re` library and remove these values using regex.
- Subtitles often output one sentence at a time, so I use sentence breaking to separate the sentences. I utilize the existing sentence structure provided by the subtitles themselves.
- Punctuation marks are then removed using the NLTK library.
- I attempted another method for sentence tokenization using the `sent_tokenize` NLTK function, but the results were not satisfactory. Therefore, I preferred to rely on the sentence structure provided by the subtitles.
- For word tokenization, I used the `word_tokenize` function from the NLTK library.

5 Structure of cleaned data

The structure of the cleaned data is as follows: all the data is stored within the "clean" directory. Each subtitle is cleaned from the "raw" folder and saved in the "clean" folder as a TXT file with its corresponding ID as the filename.

Then, I bring the cleaned data into pandas data frames, where each data frame consists of a list of sentences from each subtitle, along with their respective labels. These data frames are saved in the "sentencebroken" folder. Additionally, a separate data frame is created for each subtitle, containing a list of words along with their corresponding labels. These data frames are saved in the "wordbroken" folder. Finally, these data frames are saved as CSV files. These CSV files uploaded to Hugging Face for further processing.

6 Reports

The following reports are presented in the form of tables and charts, providing information about the data.

General Report

For each label we have :

- Number of subtitles (data)
- Numebr of sentences
- Number of words
- Number of unique words (non-duplicate words)

label	number of data	number of sentences	number of words	number of unique words
NC-17	18	27649	120356	8322
PG-13	344	681435	3061467	51651
G	77	141108	633729	21846
R	602	1126234	5074263	58495
PG	413	756838	3363372	48769
All	1454	2733264	12253187	96241

Table 1: Genral Report

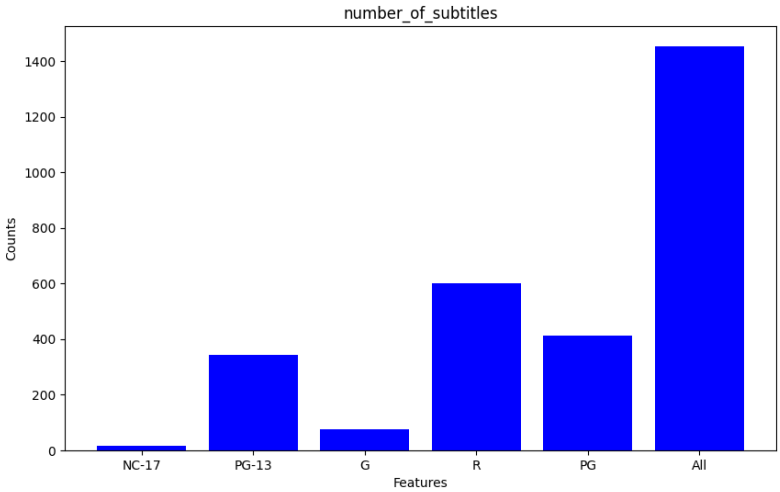

Figure 1: Number of subtitles

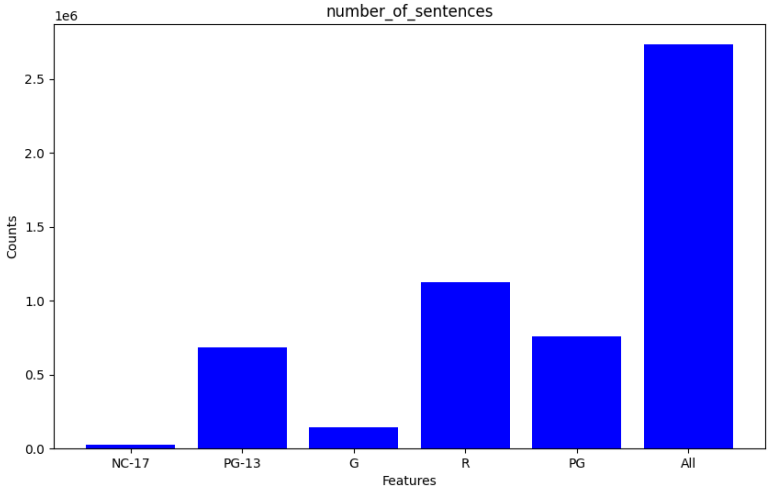

Figure 2: Number of sentences

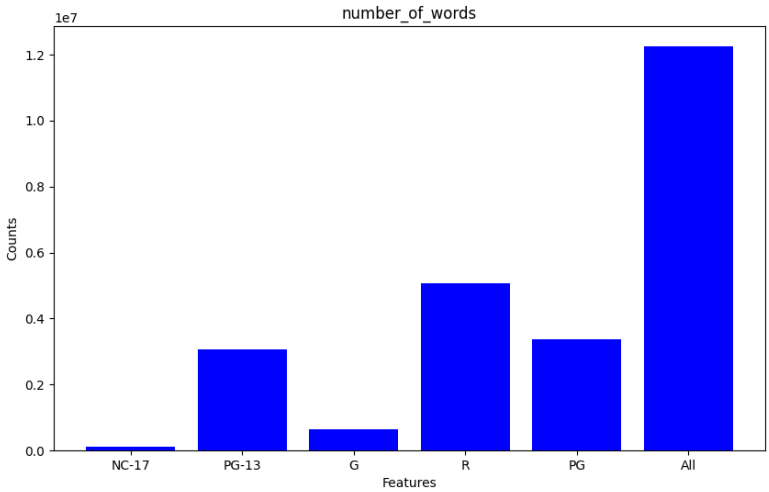

Figure 3: Number of words

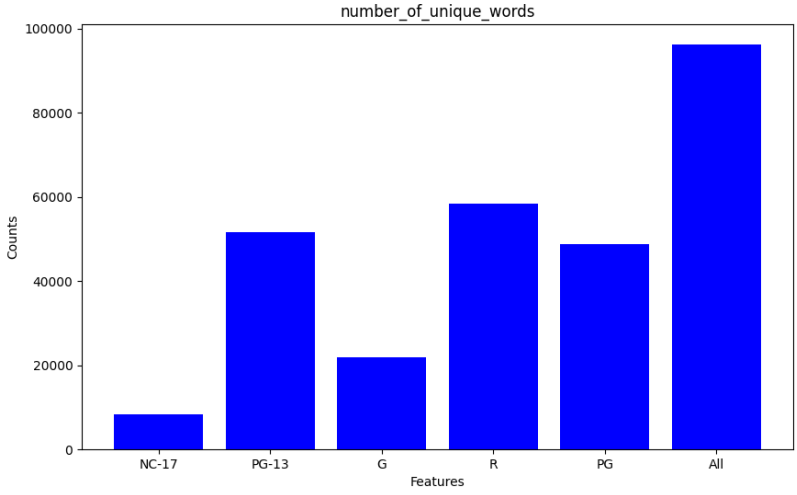

Figure 4: Number of unique words (non-duplicate)

Number of unique words that are common and non-common in every two labels

Labels	Number of common tokens
PG/R	31163
R/PG	31163
PG/G	16479
G/PG	16479
PG/PG-13	29123
PG-13/PG	29123
PG/NC-17	7323
NC-17/PG	7323
R/G	16632
G/R	16632
R/PG-13	31789
PG-13/R	31789
R/NC-17	7540
NC-17/R	7540
G/PG-13	16439
PG-13/G	16439
G/NC-17	5902
NC-17/G	5902
PG-13/NC-17	7315
NC-17/PG-13	7315

Table 2: Common words

Labels	Number of non-common tokens
PG/R	17606
R/PG	27332
PG/G	32290
G/PG	5367
PG/PG-13	19646
PG-13/PG	22528
PG/NC-17	41446
NC-17/PG	999
R/G	41863
G/R	5214
R/PG-13	26706
PG-13/R	19862
R/NC-17	50955
NC-17/R	782
G/PG-13	5407
PG-13/G	35212
G/NC-17	15944
NC-17/G	2420
PG-13/NC-17	44336
NC-17/PG-13	1007

Table 3: Non-common words

10 non-common words of each label

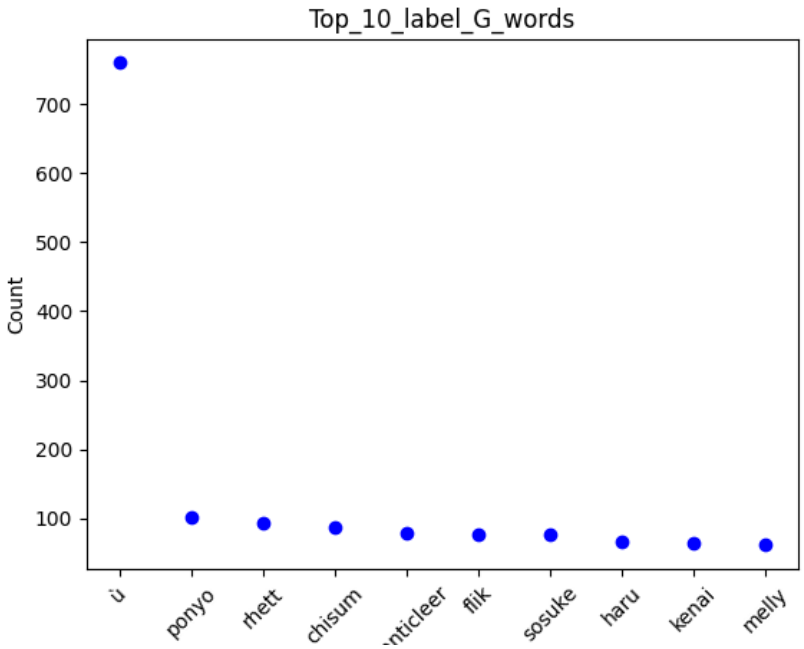

Figure 5: Top 10 label G words

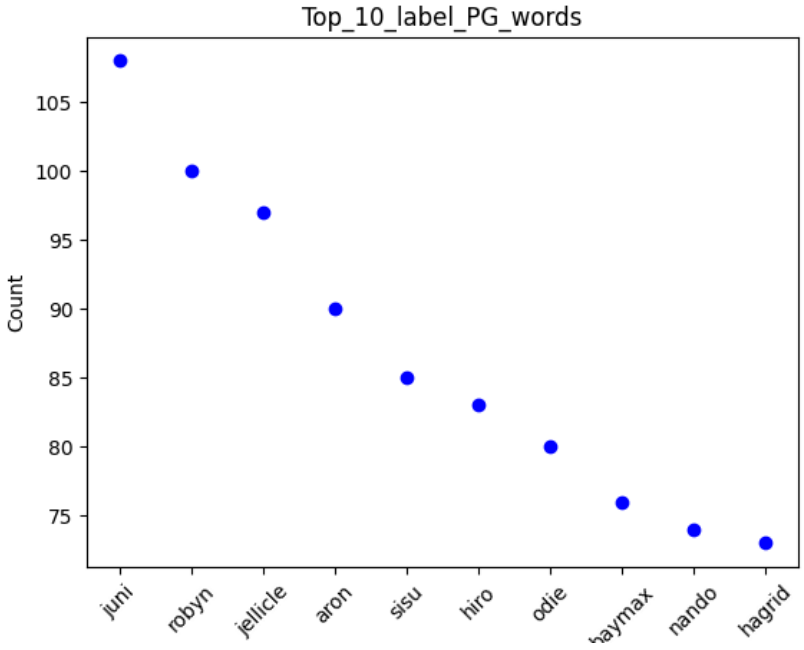

Figure 6: Top 10 label PG words

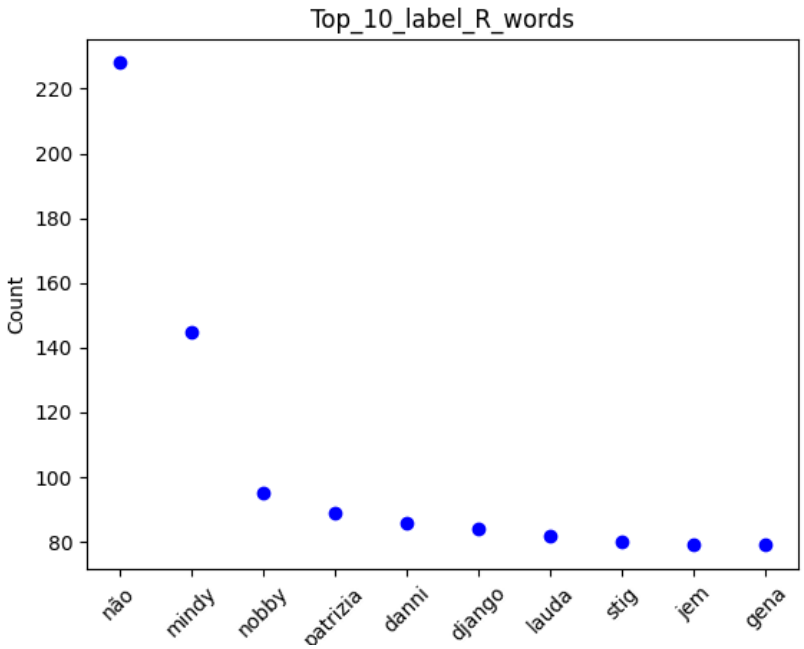

Figure 7: Top 10 label R words

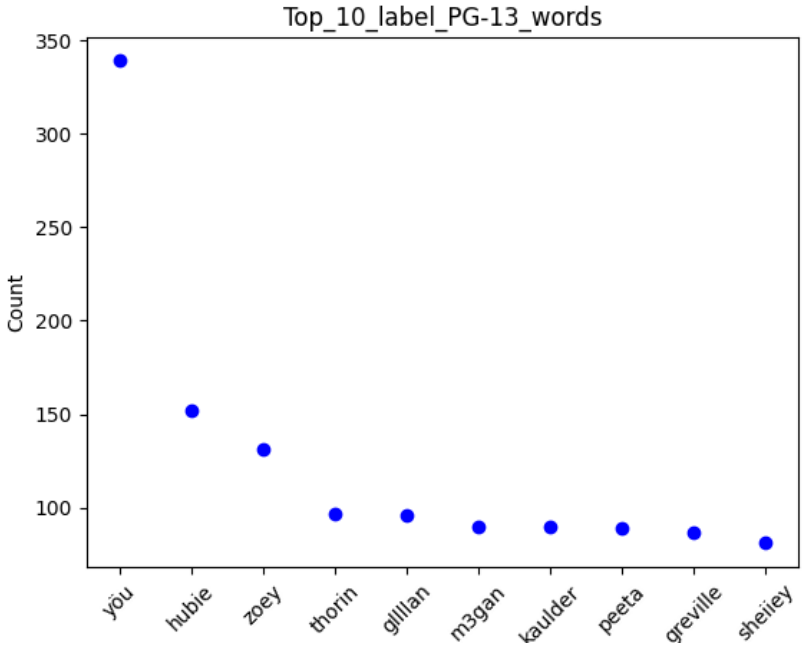

Figure 8: Top 10 label PG-13 words

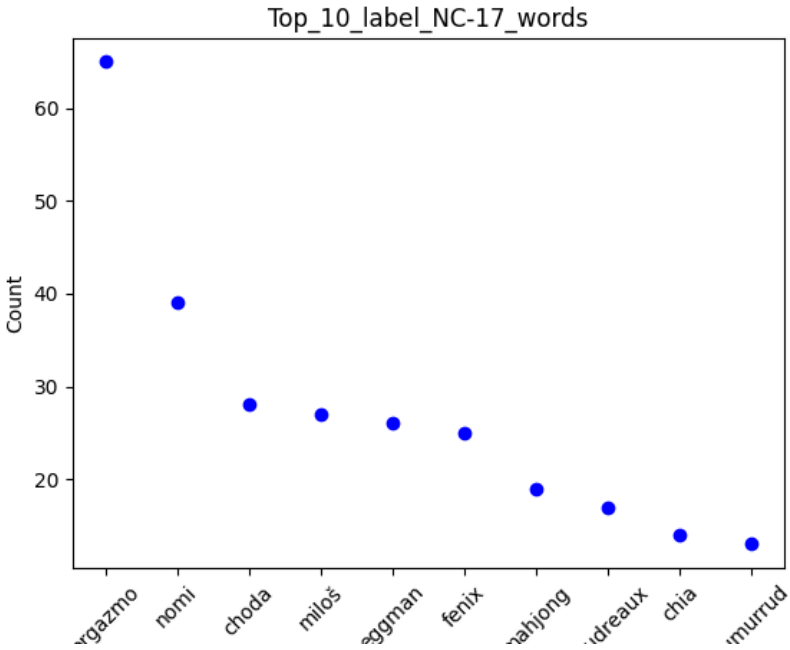

Figure 9: Top 10 label NC-17 words

The top 10 common words for each label compared to other labels based on the relative normalized frequency criterion.

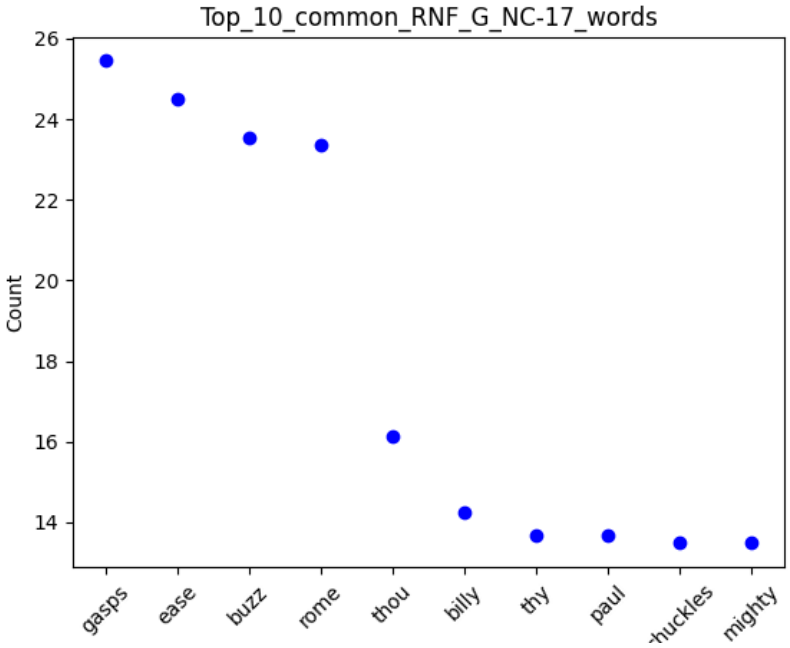

Figure 10: RNF G NC-17

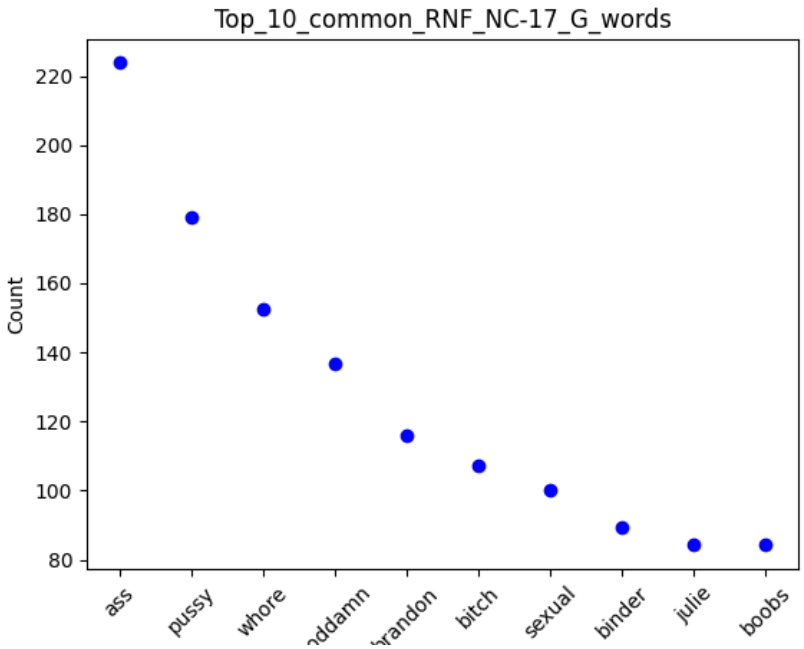

Figure 11: RNF NC-17 G

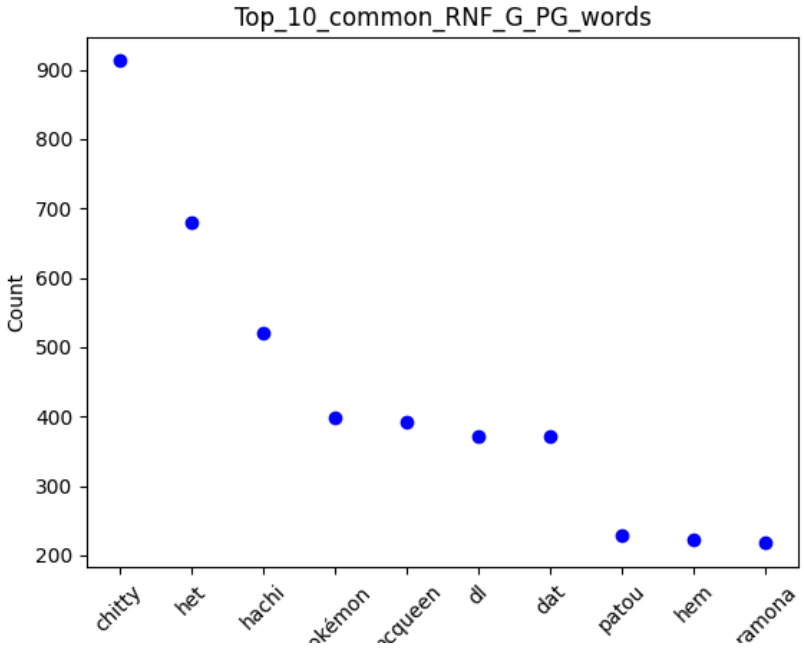

Figure 12: RNF G PG

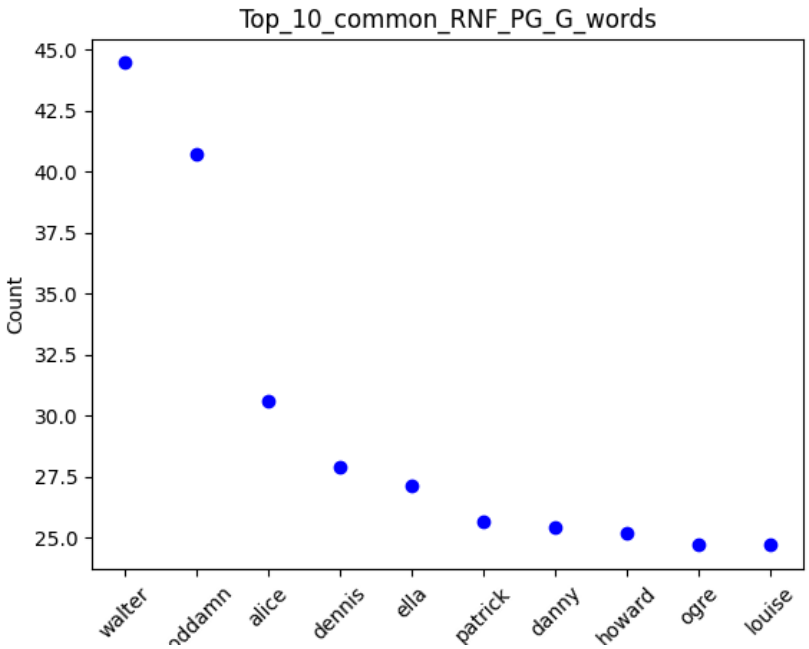

Figure 13: RNF PG G

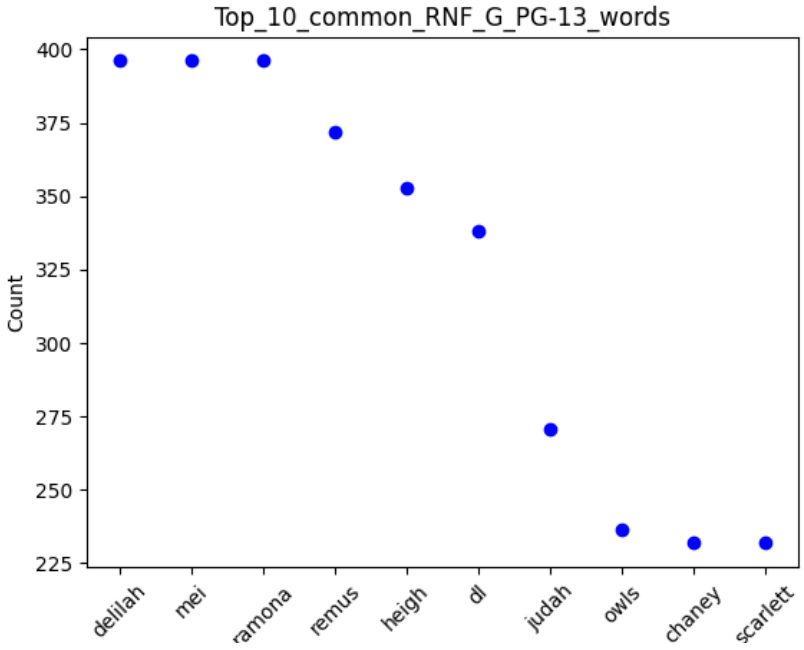

Figure 14: RNF G PG-13

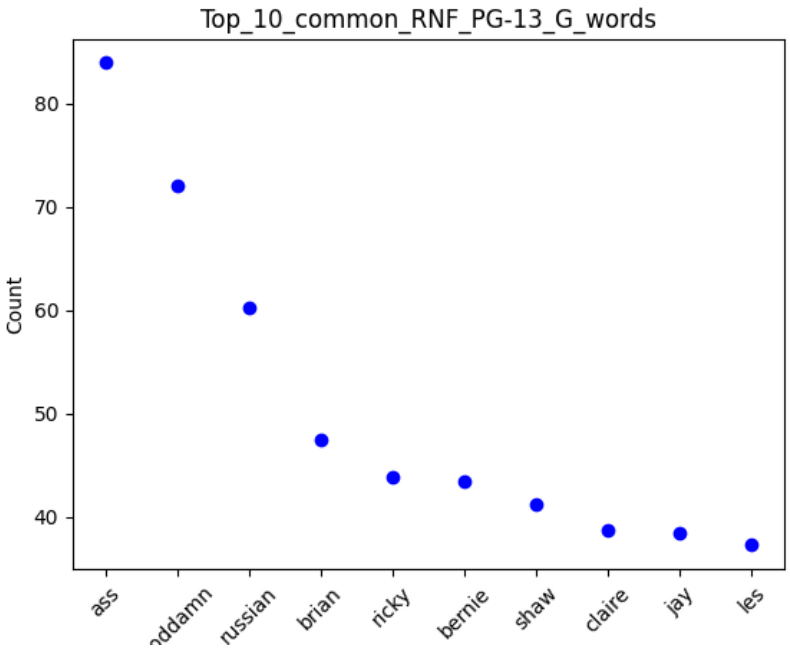

Figure 15: RNF PG-13 G

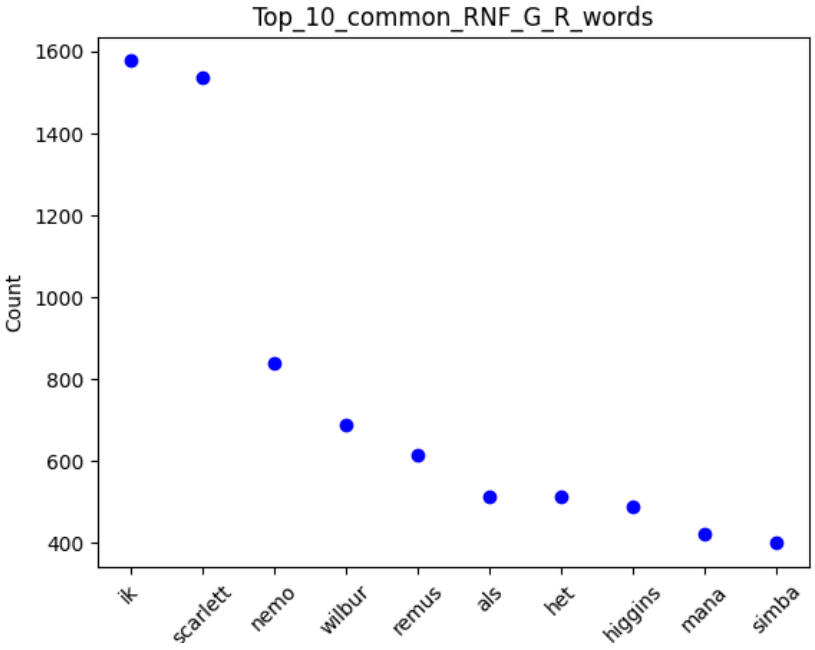

Figure 16: RNF G R

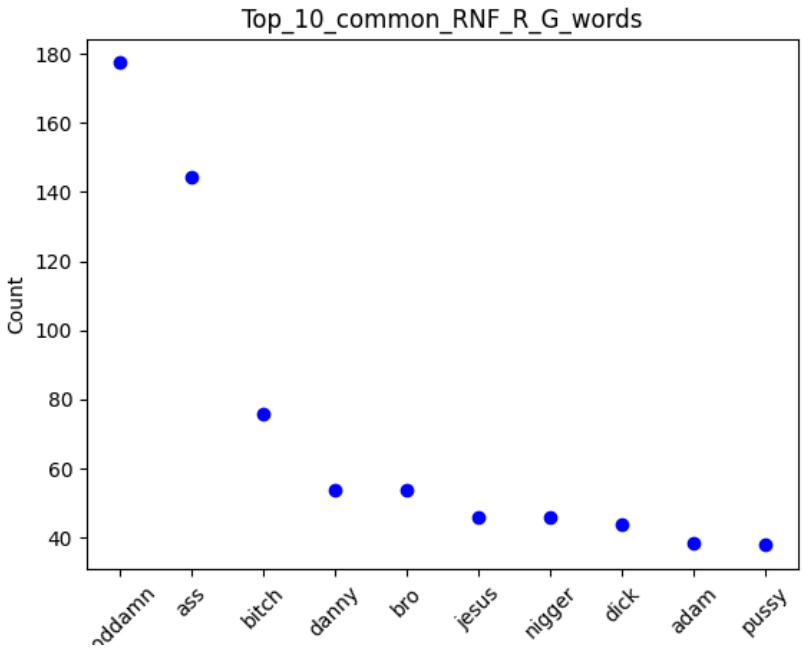

Figure 17: RNF R G

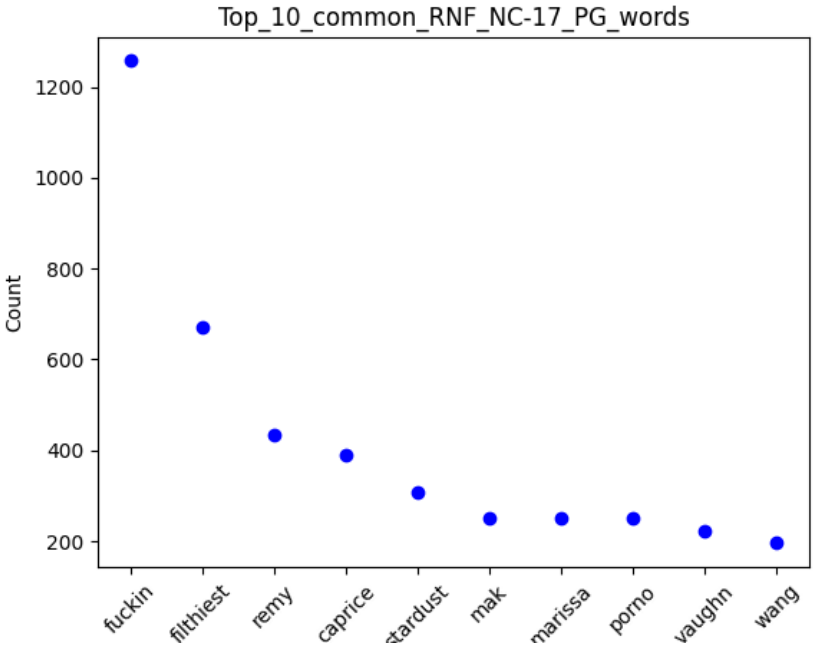

Figure 18: RNF NC-17 PG

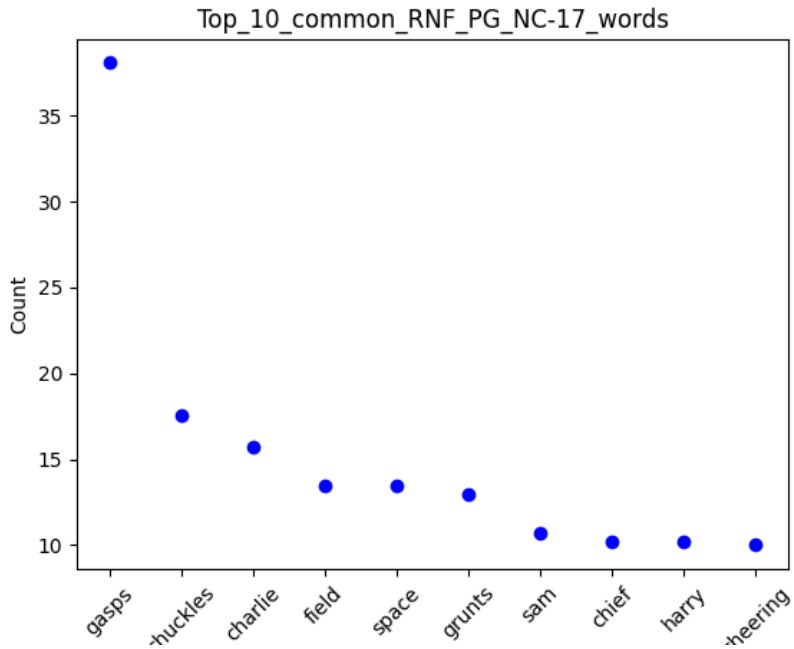

Figure 19: RNF PG NC-17

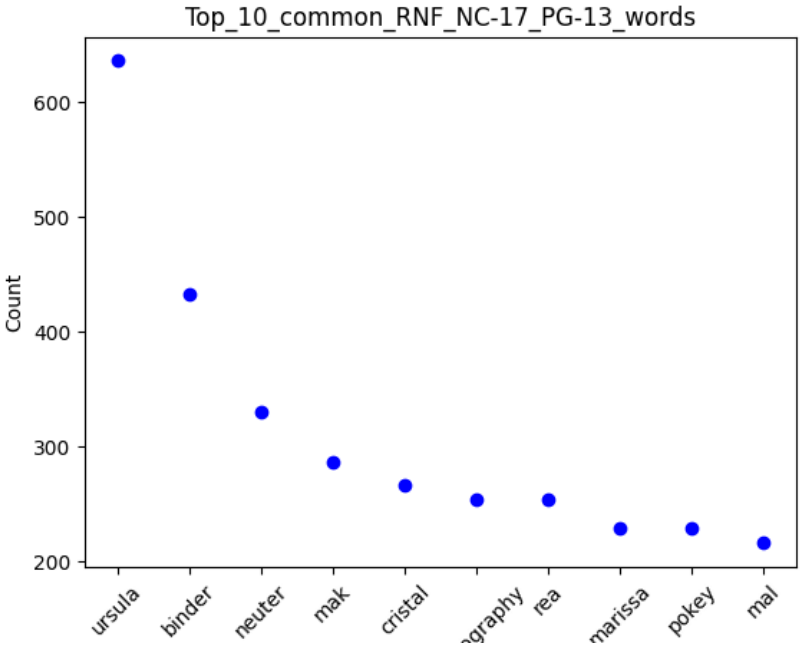

Figure 20: RNF NC-17 PG-13

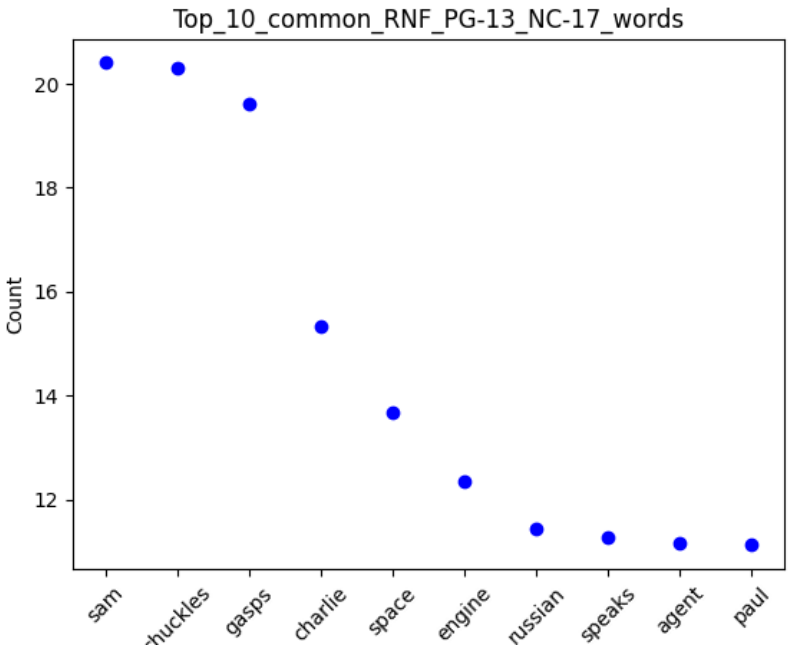

Figure 21: RNF PG-13 NC-17

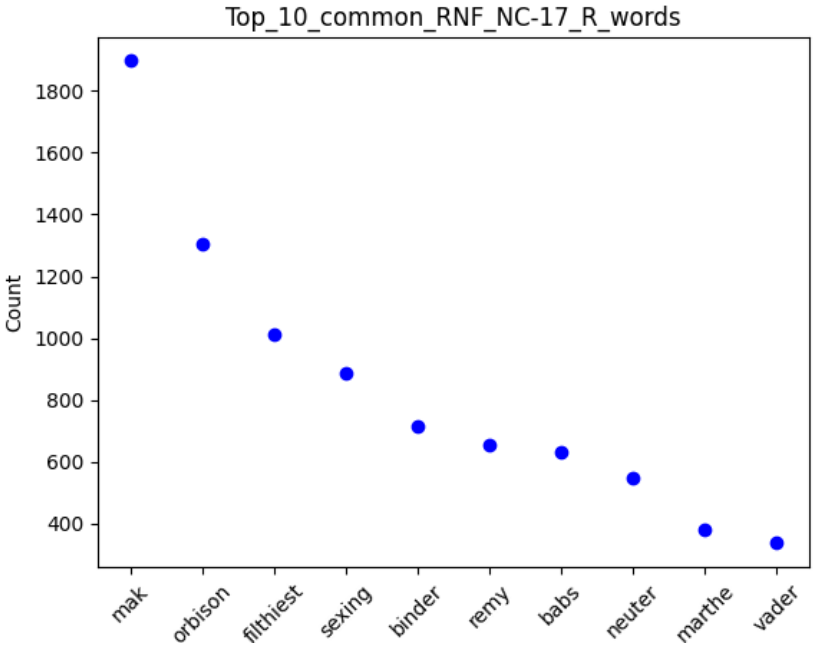

Figure 22: RNF NC-17 R

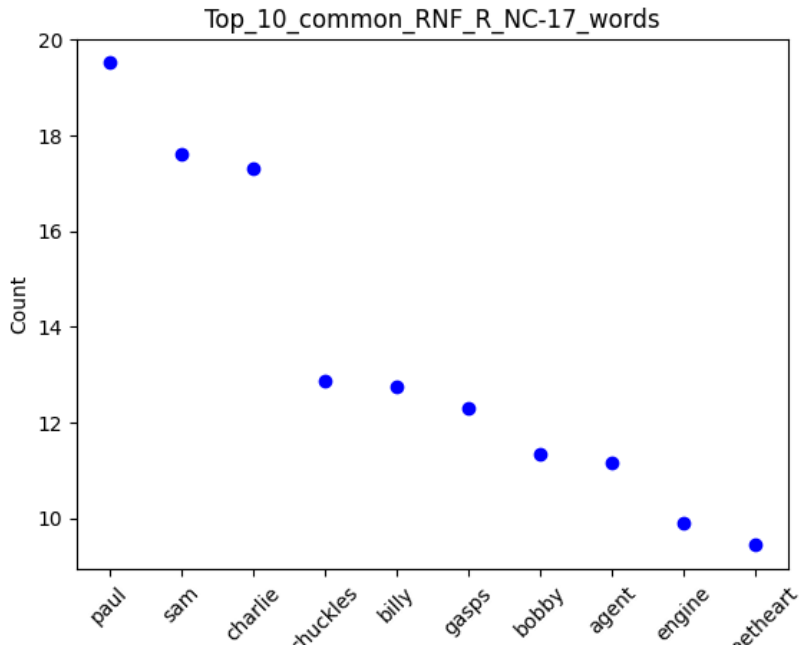

Figure 23: RNF R NC-17

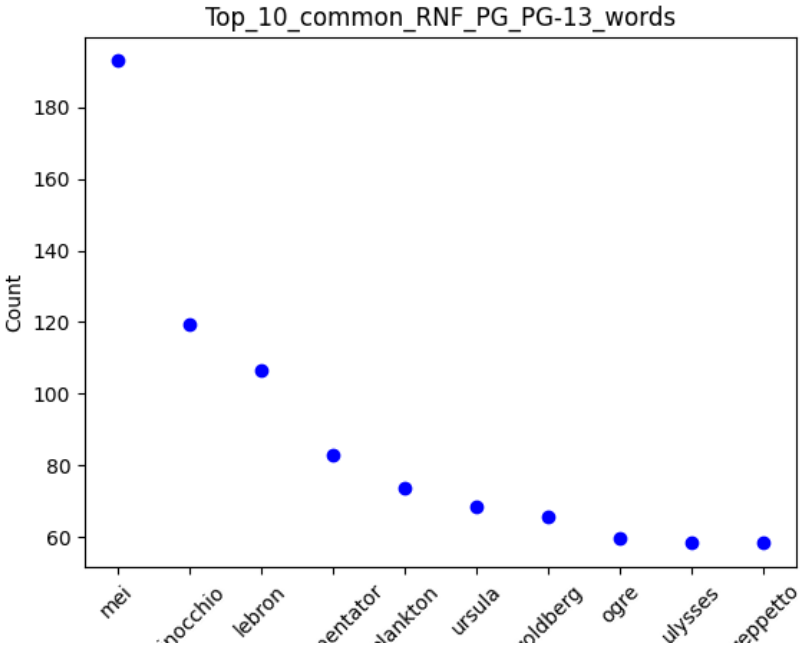

Figure 24: RNF PG PG-13

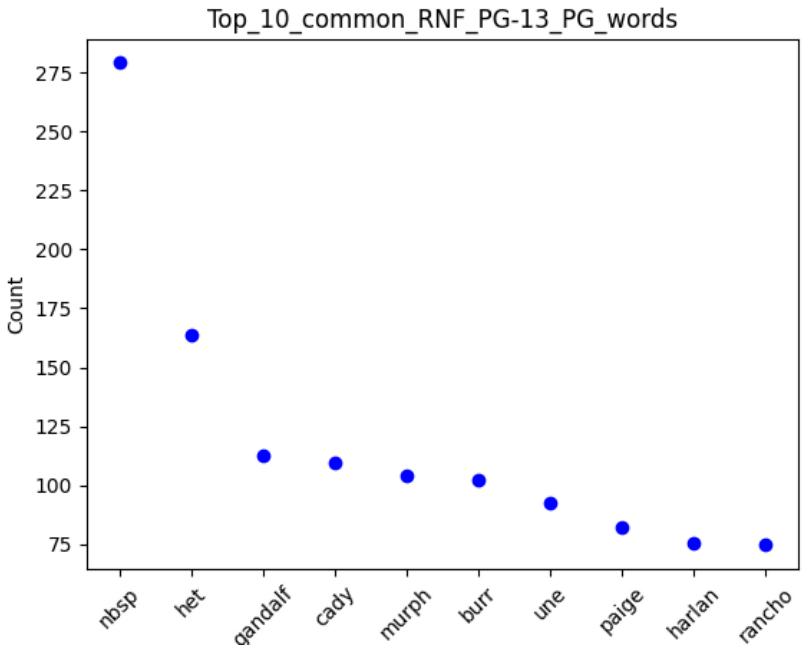

Figure 25: RNF PG-13 PG

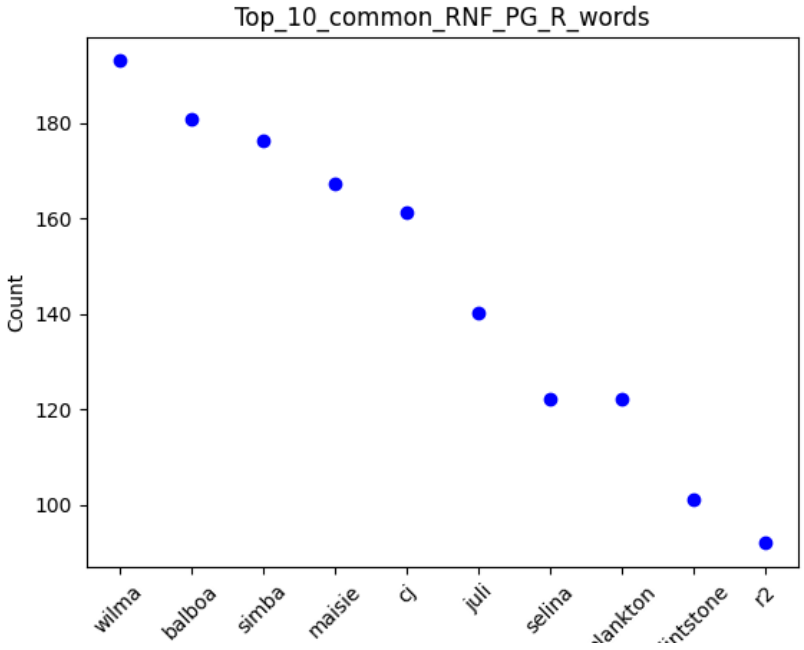

Figure 26: RNF PG R

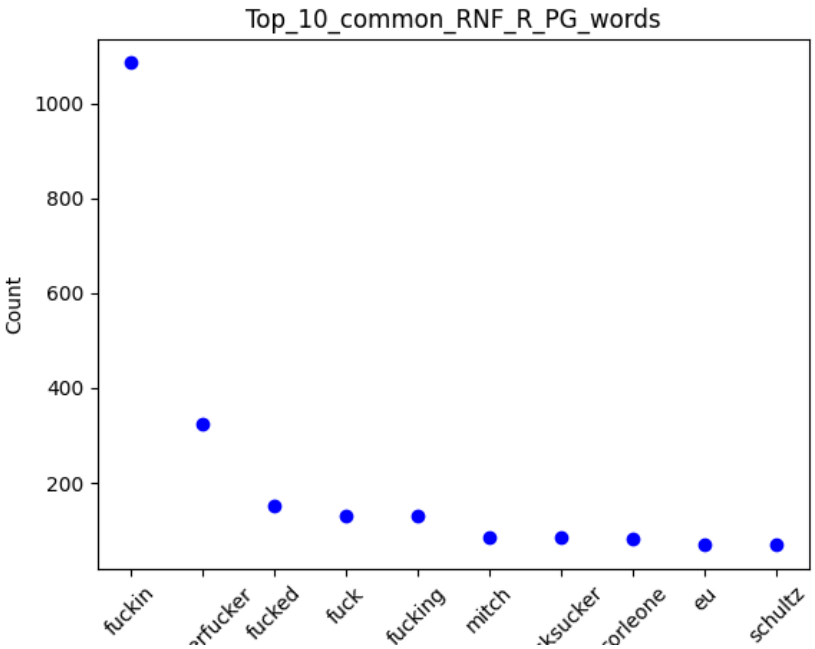

Figure 27: RNF R PG

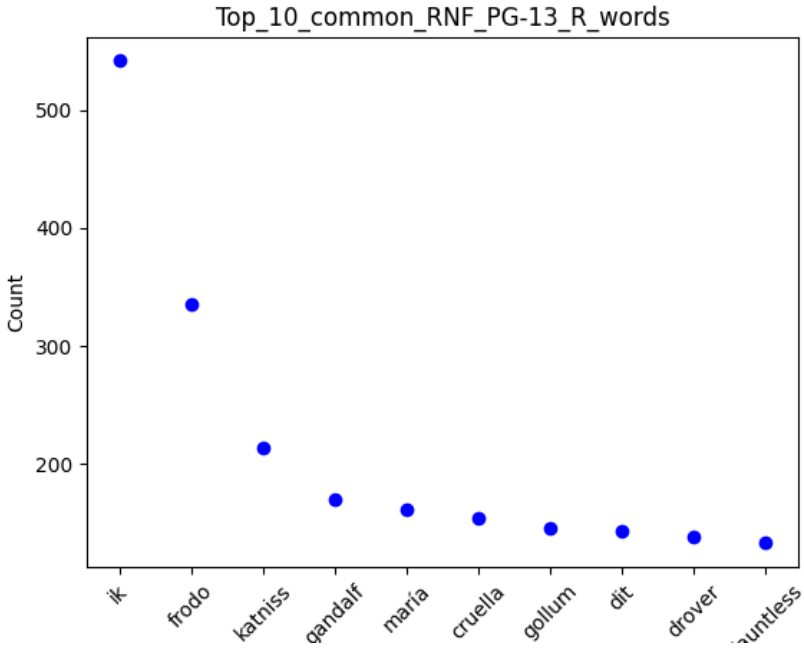

Figure 28: RNF PG-13 R

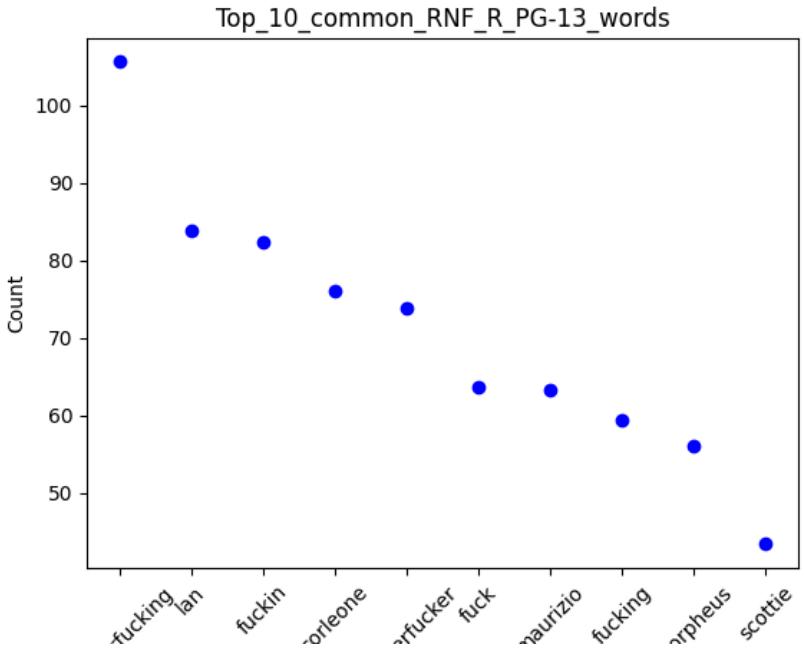

Figure 29: RNF R PG-13

Top 10 Words for each label based of TF-IDF

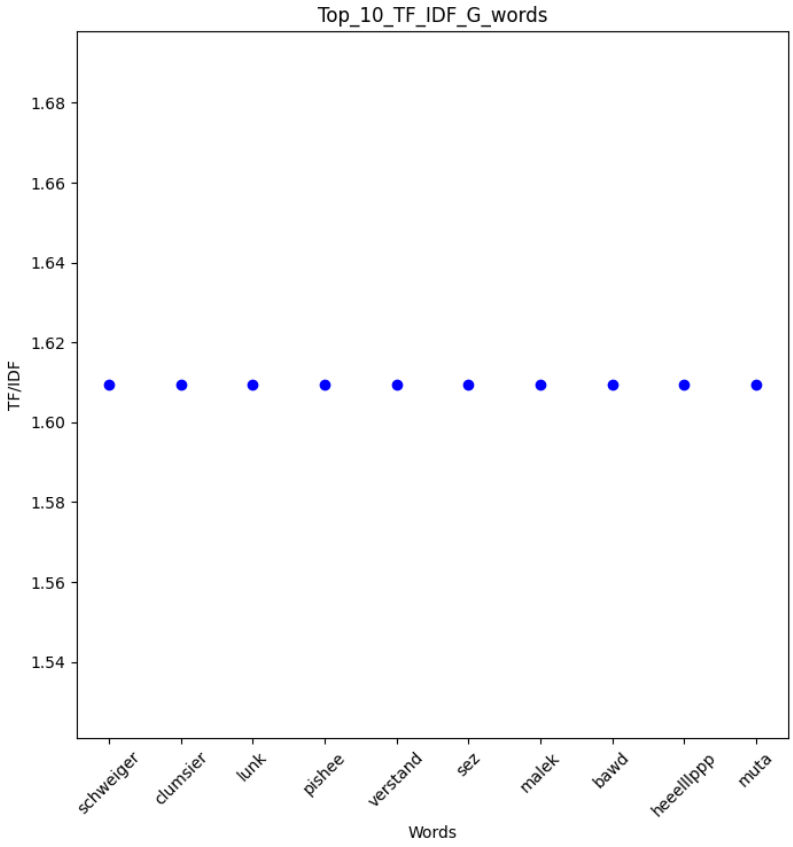

Figure 30: Top 10 TF-IDF G words

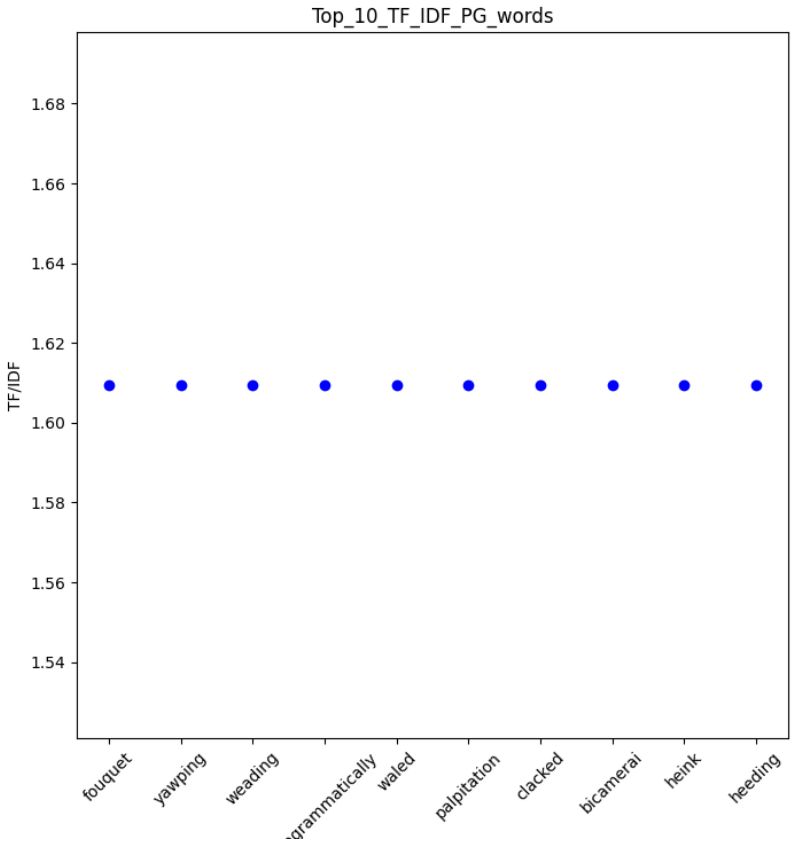

Figure 31: Top 10 TF-IDF PG words

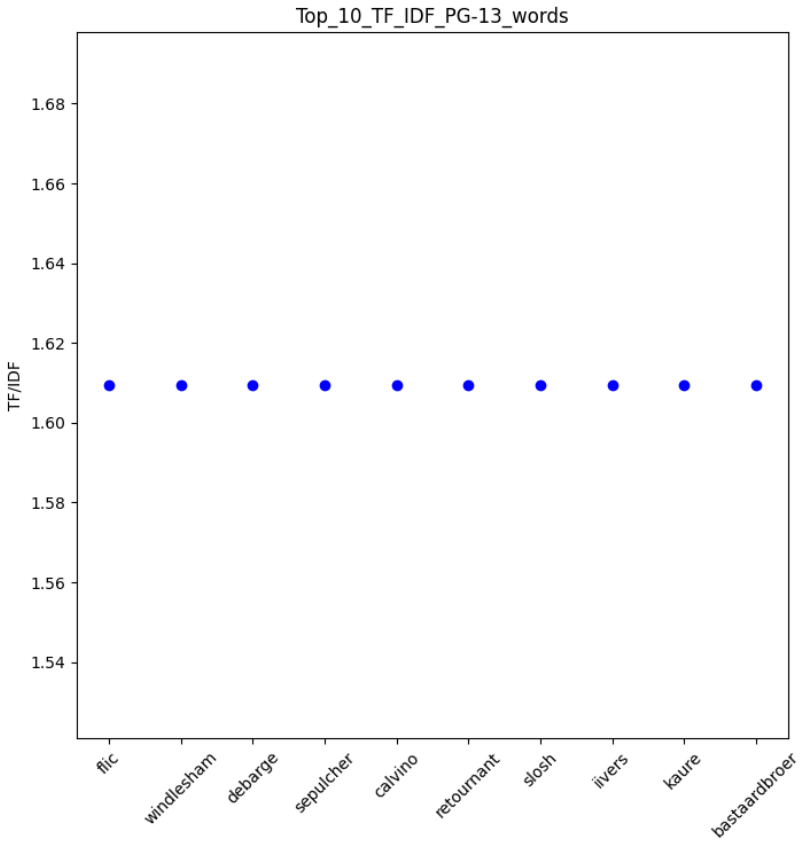

Figure 32: Top 10 TF-IDF PG-13 words

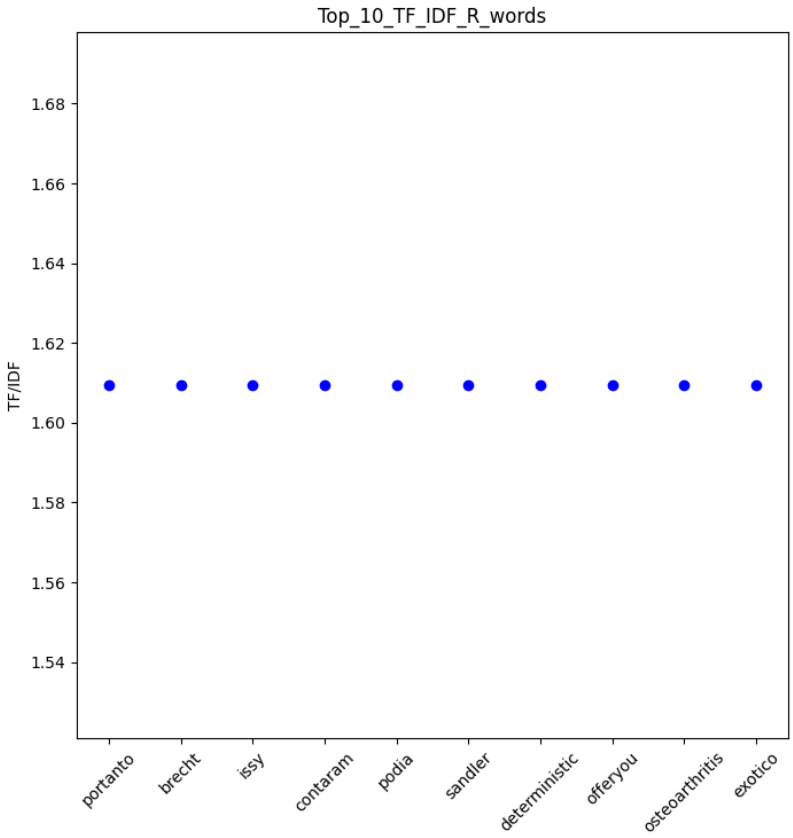

Figure 33: Top 10 TF-IDF R words

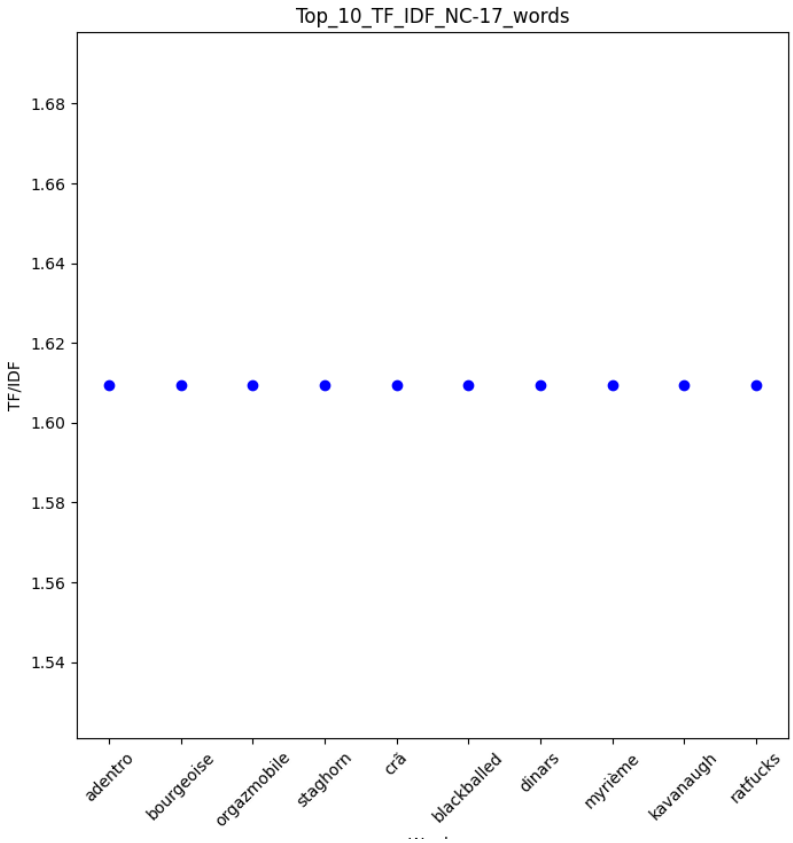

Figure 34: Top 10 TF-IDF NC-17 words

Top 15 Words for each label histogram (from high to low freq)

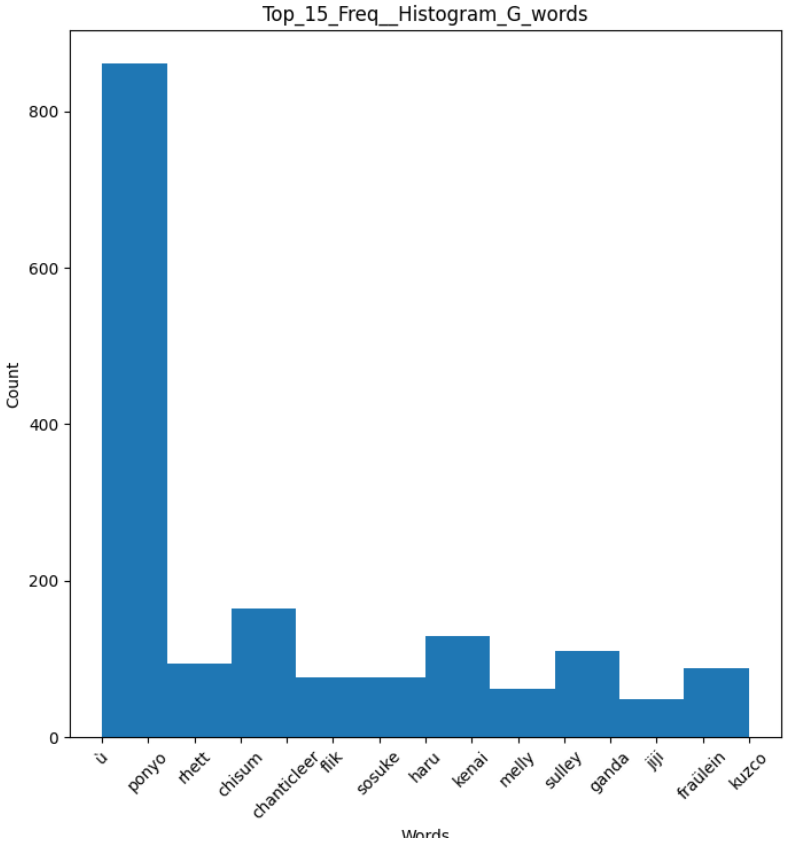

Figure 35: Top 15 Freq Histogram G Words

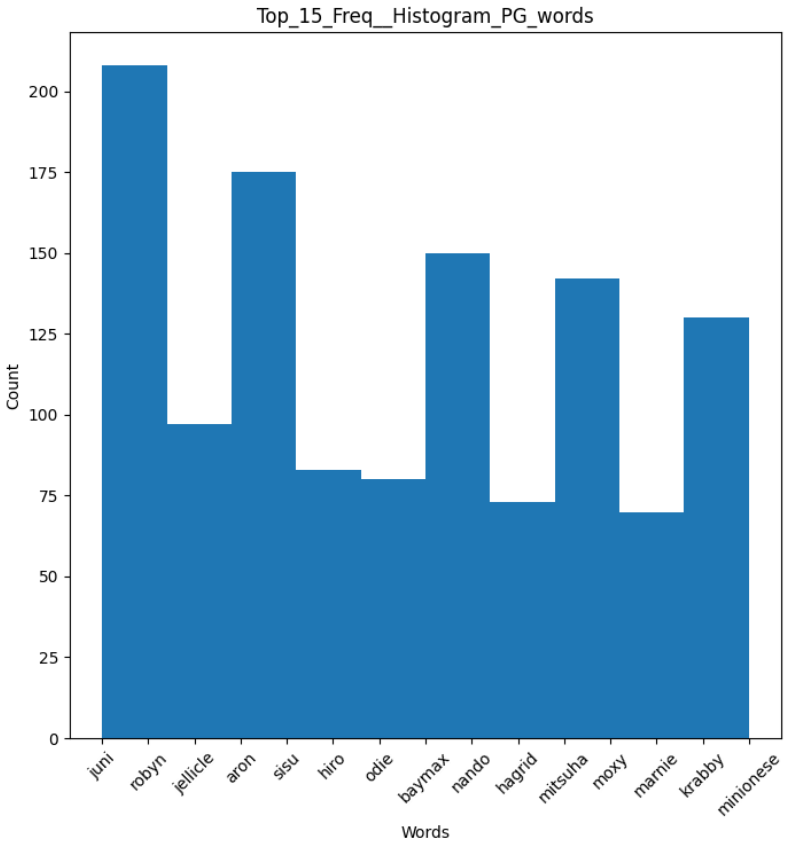

Figure 36: Top 15 Freq Histogram PG Words

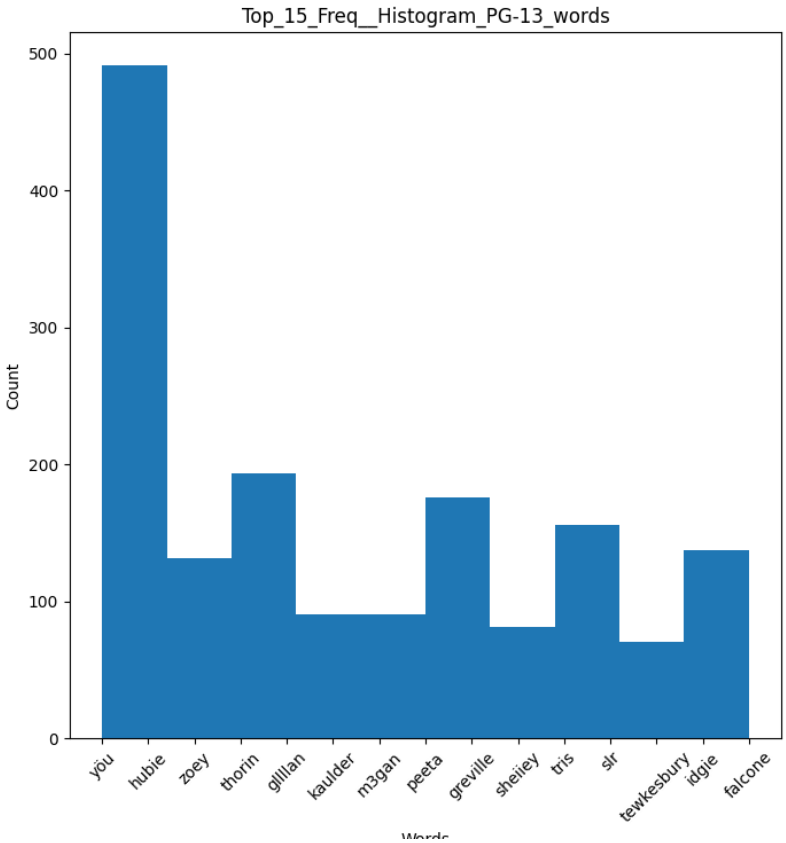

Figure 37: Top 15 Freq Histogram PG-13 Words

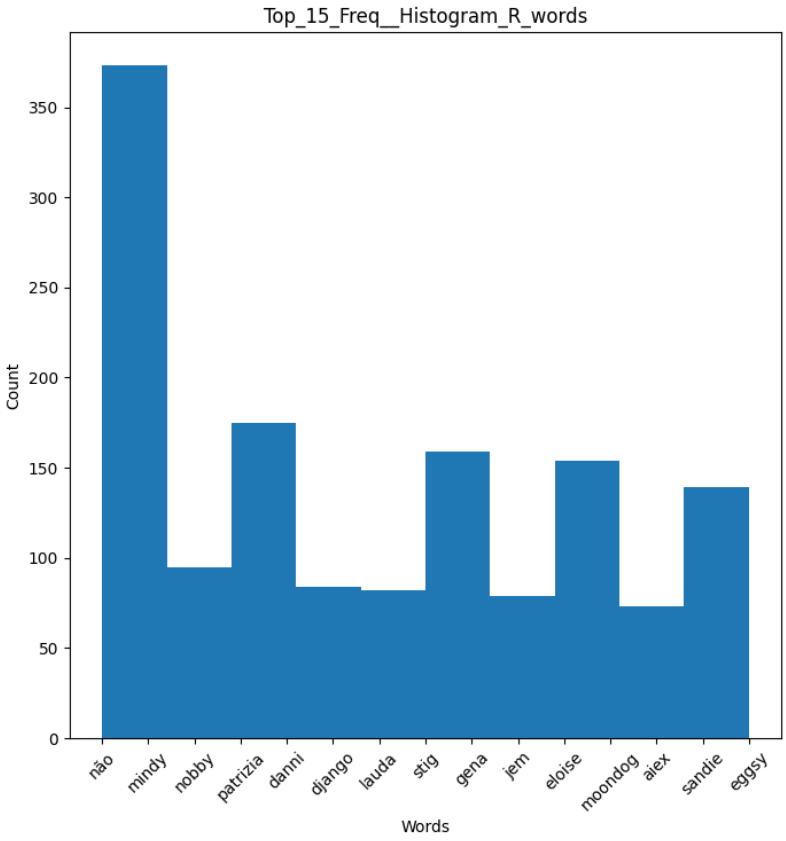

Figure 38: Top 15 Freq Histogram R Words

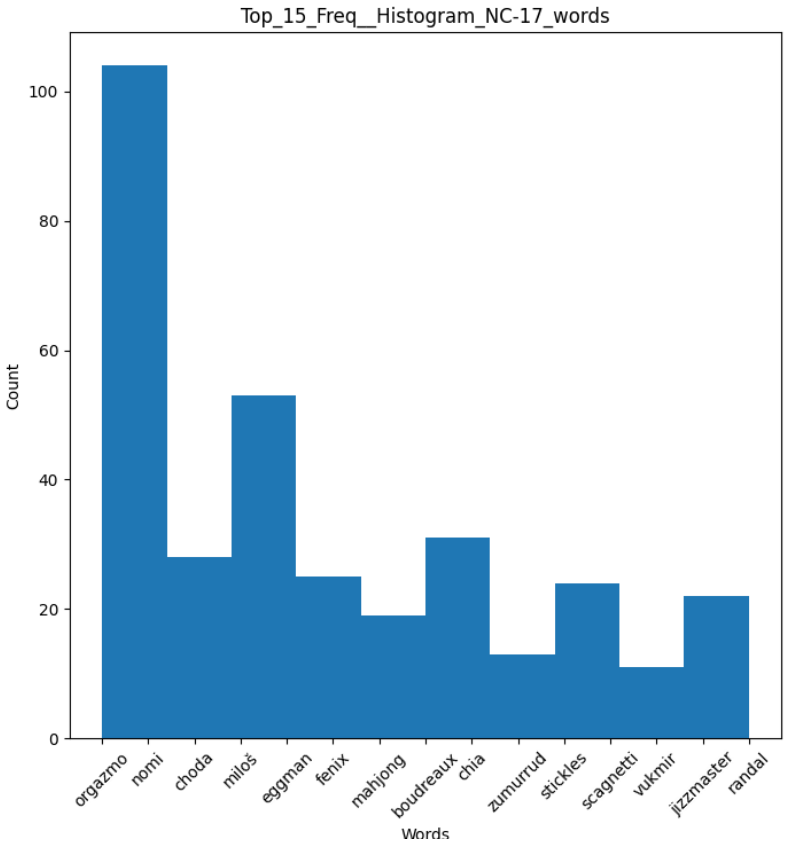

Figure 39: Top 15 Freq Histogram NC-17 Words