2024

Big Data Analytics and Social Media

Derek Qiu

7230ICT

2024-4-30

Contents

1. Introduction

As the manager of the illustrious Lady Gaga, I am constantly exploring innovative strategies to enhance her already formidable presence in the entertainment industry.

Our objective is clear: to leverage social media analytics to unlock new dimensions of popularity and engagement. By meticulously analyzing interactions and trends across platforms, we aim to optimize our promotional strategies and foster deeper connections with fans. Through this endeavor, we will harness the power of big data to propel Lady Gaga's artistic vision further into the global cultural zeitgeist, ensuring her enduring influence and relevance in an ever-evolving digital landscape. This case study will detail our approach, from data collection to strategic implementation, highlighting the transformative potential of analytics in enhancing the stature of one of the world's most dynamic performers.

1.1 Artist Background

Lady Gaga, born Stefani Joanne Angelina Germanotta on March 28, 1986, in New York City, is an American singer-songwriter and actress renowned for her flamboyant costumes, provocative lyrics, and robust vocal abilities (Levy, 2024). Gaga began her career in the music industry in 2008 with the release of her debut album, "The Fame," which became a global success and solidified her position in the pop culture landscape.

Lady Gaga has released a total of seven studio albums, two compilation albums, and numerous singles, amassing a discography that includes over 40 singles including popular hits such as "Just Dance," "Poker Face," and "Bad Romance" (billboard, 2023). Her influence extends beyond music into film and television, where she has also achieved significant success. A role in "A Star Is Born", for which she received an Academy Award nomination for Best Actress. This role underscored her versatility and depth as a performer, showcasing her ability not only to entertain but also to connect with audiences on a profound level (Montalti, 2022).

Furthermore, Lady Gaga is also a prominent advocate for LGBTQ+ rights and mental health awareness, using her platform to support various causes and inspire a vast fan base worldwide (Wong, 2024). Her ability to continuously reinvent her music and image while maintaining an authentic connection with fans is a testament to her enduring popularity and impact on the entertainment industry.

Lady Gaga's fashion choices often spark widespread public and critical discourse, with opinions sharply divided. Such as the meat dress worn at the 2010 MTV Video Music

Awards, sparks widespread debate about fashion, art, and celebrity culture (Denise Winterman, Jon Kelly, 2010). Some admire her aesthetic as groundbreaking and artistically profound, while others criticize it as mere attention-seeking. However, these polarized views only serve to amplify her prominence in the realm of pop culture, where she remains a figure of substantial influence and debate.

In her 16 years in the limelight, Lady Gaga has not only entertained but also provoked thought, making her one of the most influential figures in contemporary pop culture. Her work challenges conventions and makes bold statements, reflecting her mastery of the art of celebrity.

2. Data Selection and Exploration

2.1 Data Collection

For this analysis, I chose "Alejandro," a song by Lady Gaga on YouTube known for its controversial themes and imagery. Additionally, I selected the very popular song "Poker Face" as a comparison to "Alejandro" to analyze the differences in public reception between the two songs and identify areas for Lady Gaga to improve and gain more popularity from her music. Furthermore, I chose a Reddit thread about her iconic flesh skirt, which was a topic that generated a lot of discussion and media attention. These resources were selected for their potential to provide in-depth public perspective on the impact of Lady Gaga's art and public statements.

2.1.1 YouTube

For the YouTube data collection, I used the YouTube API to extract comments from the videos. Specifically, I collected 3,000 comments from the "Poker Face" video and 3,000 comments from the "Alejandro" video. The code snippet below demonstrates the process:

```
# Authenticate to YouTube and collect data for Alejandro
my_api_key <- "AIzaSyBATjeVOu2-RuNDnlukfa3v25fIsK5GzPI"
yt_auth <- Authenticate("youtube", apiKey = my_api_key)
alejandro_video_url <- c("https://www.youtube.com/watch?v=niqrrmev4mA")

alejandro_yt_data <- yt_auth |> Collect(videoIDs = alejandro_video_url,
                                        maxComments = 3000,
                                        writeToFile = TRUE,
                                        verbose = TRUE)

# View collected YouTube data for Alejandro
View(alejandro_yt_data)
```

Figure 1R code of fetching Alejandro youtube video

	Comment	AuthorDisplayName	AuthorProfileImageUrl	AuthorChannelUrl
1	Каждый раз, как в первый бегут мурашки.. Обожаю! ❤	@sophie_2307	https://yt3.ggpht.com/ytc/AIdro_loEDtFUh7wHmHasHxCHu...	http://www.youtube.com/@s
2	Klipin ortasında Maral Tahirqızını gözləyənlər😍	@Gasimova155	https://yt3.ggpht.com/y_N_bs1K1h_B_VKXmNJQ1ZJWzsjjmc...	http://www.youtube.com/@C
3	Gaga c'est une très belle femme ❤	@ChafaDjamel	https://yt3.ggpht.com/ytc/AIdro_ko1bohorF43L7N2__iJfOUy...	http://www.youtube.com/@C
4	Я знову психічно урівноважена	@muriance	https://yt3.ggpht.com/ytc/AIdro_nqqp5mNy5Z8eelc-bQVBg...	http://www.youtube.com/@n
5	A masterpiece	@Independent5159	https://yt3.ggpht.com/ytc/AIdro_nTCRmCpUsgUhrwCn3NSg...	http://www.youtube.com/@I
6	This is the times when Gaga has weirdest MV that people th...	@JamieBentall	https://yt3.ggpht.com/kaoU-3GTyVWyee9QtQYarYtynxHGeE...	http://www.youtube.com/@J
7	2090 Anyone?!! 😳	@2004fog	https://yt3.ggpht.com/QDQMh4TxoYdf7rn0300alD6p_lYIA8...	http://www.youtube.com/@2
8	悪魔崇拝者?	@user-fy5rf4gw7o	https://yt3.ggpht.com/ytc/AIdro_lvM2js-8QP4Tz3x2lnVd5_P...	http://www.youtube.com/@u
9	❤	@Jiswealth	https://yt3.ggpht.com/ytc/AIdro_mO4NT9doXyYXiQpZI84S...	http://www.youtube.com/@J
10	2024 and I love that even more	@pw1244	https://yt3.ggpht.com/ytc/AIdro_lImWjpMrlvdRovD-M6t404...	http://www.youtube.com/@p
11	A música faz 13 anos atrás, eu era criança com 6 anos e até ...	@cesarmamede2178	https://yt3.ggpht.com/HfeEyOUExYXWkDLLz40vfs_NQTHfS_...	http://www.youtube.com/@c
12	Still savoring these sexy sounds baby!!! 😊🙏	@vanessamorey3812	https://yt3.ggpht.com/ytc/AIdro_nJvNo4ZjUz2aDXZ-5HxYXE...	http://www.youtube.com/@v
13	Anyone in 2024 may?	@qLeviz0	https://yt3.ggpht.com/YCteYKrwzW2O0UJUgx9B70rXUxjW3...	http://www.youtube.com/@q
14	Anyone named Fernando like me? Anyone?	@Royisthegoat_539	https://yt3.ggpht.com/09-BSEFXYnGja5Wqob40WnDns7S7K...	http://www.youtube.com/@R
15	Alejandro is dewan @comeoverwjenyouresober	@rahulprasad2162	https://yt3.ggpht.com/ytc/AIdro_mItE3dgt8YFPiKy7e8_6648...	http://www.youtube.com/@r
16	Andjelkovic Dennis : die Frauen wollten doch sehen was pur...	@LPRpgler25	https://yt3.ggpht.com/8F11lYVYlsqZUj5dfgdBTqBHI0kwssM...	http://www.youtube.com/@L
17	This video still weirded tf out of me 🤯 the satanic ritual wa...	@ean1989tv	https://yt3.ggpht.com/eZoew7lsoN7WZg-PtuBArLj_CMv7xm...	http://www.youtube.com/@e
18	MV 13 năm trước mà đỉnh vl 🤯 that's fanatic mv still in 2024	@teddy19194	https://yt3.ggpht.com/A8rnDuz1i6o-DsiqHJOApg7tNwevlZA...	http://www.youtube.com/@t
19	THIS IS a f*ck!g MASTERPIECE❤	@mateotorres9108	https://yt3.ggpht.com/ytc/AIdro_mzwk73fmnfDwFs4aca8l6k...	http://www.youtube.com/@n
20	Get the fuck out of my fucking face.	@fantariwegbarwatino557	https://yt3.ggpht.com/ytc/AIdro_lSxERKmuXpC-ZC9j7fDJwY...	http://www.youtube.com/@f
21	So its really 8 minutes	@mcarcel7147	https://yt3.ggpht.com/6iZ_2sVuVD1x_z4IiT_m-8Nz6Zqm7A...	http://www.youtube.com/@n

Figure 2 Result of Alejandro youtube video data

```r
# Authenticate to YouTube and collect data
my_api_key <- "AIzaSyBATjeV0u2-RuNDnlukfa3v25fIsK5GzPI"
yt_auth <- Authenticate("youtube", apiKey = my_api_key)
video_url <- c("https://www.youtube.com/watch?v=bESGLojNYSo")    #Poke Face
yt_data <- yt_auth |> Collect(videoIDs = video_url,
                              maxComments = 3000,
                              writeToFile = TRUE,
                              verbose = TRUE) # use 'verbose' to show download progress

# View collected YouTube data for Poke Face
View(yt_data)
```

Figure 3R code of fetching Poker Face youtube video

Assignment1.R | yt_data | 4-1_Lab_Script.R | 3-2_Lab_Script.R | Creating-YouTube-Network-Graphs.R | 5-1_Lab_Script.R | 2-2_Lab_Script.R | 4-2_Lab_Script.R

Filter

	Comment	AuthorDisplayName	AuthorProfileImageUrl	AuthorChannelUrl
1	VAN DER LINDE VAN DER LINDE 🗣🗣!?	@callahann778	https://yt3.ggpht.com/5wjlRxDPlqu0Wl_tuxq58FryyBhaAgRY...	http://www.youtube.com/@callahann778
2	Alguien 2024?	@Natali.Xiomara	https://yt3.ggpht.com/uPj89v01xk4aIlm2sK8EHn0n9Ce_ZR9...	http://www.youtube.com/@Natali.Xiomara
3	Seduction is a bitch	@Steve-yc9gk	https://yt3.ggpht.com/ytc/AIdro_nuNIfceTDlBSTQUhxQ56Y...	http://www.youtube.com/@Steve-yc9gk
4	2024 y sigo amando está hermosa canción ❤	@marcoamarcelino3632	https://yt3.ggpht.com/ytc/AIdro_kc-iBry_MZpZhJMOcsILX8e...	http://www.youtube.com/@marcoamarcelino3632
5	Vim pelo flash ⚡💥	@DEIV-gu5gf	https://yt3.ggpht.com/etV1meKi4FqaKFLOdaZFPP4VaN7P0L...	http://www.youtube.com/@DEIV-gu5gf
6	2024 ❤	@Akashmj888	https://yt3.ggpht.com/OpMfjq2WV064q_zTRV9AM5OOt95f...	http://www.youtube.com/@Akashmj888
7	This MV confirmed my Gayness 😂	@oswa2702	https://yt3.ggpht.com/ytc/AIdro_mnzr0dwwOjF9Ci-pFZKEN...	http://www.youtube.com/@oswa2702
8	Barry Allen the flash and Cisco and Caitlin 🕺🕺🧍‍♀️💃💃	@ULTRA_OUSSAMA	https://yt3.ggpht.com/LHMzoG6e7HNvqzdGHNCZqU0jOp7...	http://www.youtube.com/@ULTRA_OUSSAMA
9	Nice	@Justin-nb9tk	https://yt3.ggpht.com/3tkm3i5vCV0-nHru8Ko1df3pzQ9huu...	http://www.youtube.com/@Justin-nb9tk
10	Haram sekali	@Multazam-cf6eh	https://yt3.ggpht.com/DvdkQLcLo8fpo1xNhqz4oIjaLNwqOE...	http://www.youtube.com/@Multazam-cf6eh
11	Nostalgia 😂.	@sandraaraujo9305	https://yt3.ggpht.com/ytc/AIdro_mKinTYhGgtyd9Q25mY6w...	http://www.youtube.com/@sandraaraujo9305
12	In the flash 2024?❤	@Jasonofficial444	https://yt3.ggpht.com/rwZCZLqh_VCPG2PuoSEBOMJXvAUh...	http://www.youtube.com/@Jasonofficial444
13	Came here after watching Christopher Walken read poker fa...	@bakilacat1	https://yt3.ggpht.com/ytc/AIdro_nVyA3TgFOp6012mn7tqex...	http://www.youtube.com/@bakilacat1
14	May, 18, 2024.)	@user-tn3cg3ip7f	https://yt3.ggpht.com/ytc/AIdro_kdWDTXQfOdFmKRl2VfBJ...	http://www.youtube.com/@user-tn3cg3ip7f
15	Noose jobs don't work for me	@Spinozariahnna	https://yt3.ggpht.com/dLpk6BeAa50FbfIvctycytqci3az0E4nC...	http://www.youtube.com/@Spinozariahnna
16	For years, radio stations were playing this song uncensored, ...	@isrulius	https://yt3.ggpht.com/GrOwEys6fstjOqQ1GCgRbkiAt07sHFj...	http://www.youtube.com/@isrulius
17	Dftyd	@oscardelacruzganoza9257	https://yt3.ggpht.com/ytc/AIdro_m0R48UchWJhQCcZfYh5a...	http://www.youtube.com/@oscardelacruzganoza9257
18	Mother ❤	@JhonreyBibat	https://yt3.ggpht.com/ytc/AIdro_lT8Qu4LCD_pf9EwiAYpLW0...	http://www.youtube.com/@JhonreyBibat
19	She what her face? 🤔🤯	@jorgenoj1258	https://yt3.ggpht.com/ytc/AIdro_mBxz_tFl4TpJWMRFVzBqM...	http://www.youtube.com/@jorgenoj1258
20	She is cool 😎😃	@jhoanagutierrez2432	https://yt3.ggpht.com/ytc/AIdro_m803QGbUkfuDHVEPFdc...	http://www.youtube.com/@jhoanagutierrez2432
21	Looking back, this song is so clearly full of... well I can't even...	@soigerWhyIsThereAnAtItLooksBad	https://yt3.ggpht.com/IWLvVsnVCkV5QQ1kitM8xfAtvX_luKo...	http://www.youtube.com/@soigerWhyIsThereAnAtItLooksB
22	It's been 5 months since I'm looking for this song I have los...	@PayouteWoostensin	https://yt3.ggpht.com/2gN-j1GxFQYoQFiv6eopVLUjXgHbHv...	http://www.youtube.com/@PayouteWoostensin

Showing 1 to 22 of 3,652 entries, 12 total columns

Figure 4Result of Poker Face youtube video data

2.1.2 Reddit

For the Reddit data collection, I selected a thread discussing Lady Gaga's meat dress. I used the Reddit API to extract comments from the thread, sorted by the top comments, and collected up to 1,000 comments. The code snippet below demonstrates the process:

```
# Specify the threads from reddit which to collect data
thread_urls <- c("https://www.reddit.com/r/pics/comments/1cpfx9c/reactions_to_the_lady_gaga_meat_dress_at_the_2010/")


# Collect threads with their comments sorted by best comments first
rd_data <- Authenticate("reddit") |>
  Collect(threadUrls = thread_urls,
          sort = "TOP",
          maxComments = 2000,
          writeToFile = TRUE,
          verbose = TRUE) # use 'verbose' to show download progress

View(rd_data)
```

Figure 5R code of fetching Reddit thread

	id	structure	post_date	post_date_unix	comm_id	comm_date	comm_date_unix	num_comments	subreddit	upvote_prop	post_scor
1	1	1	2024-05-11 14:25:45	1715437545	l3l8fyo	2024-05-11 16:03:46	1715443426	49	LadyGaga	0.99	
2	2	1_1	2024-05-11 14:25:45	1715437545	l3n3odc	2024-05-11 23:31:20	1715470280	49	LadyGaga	0.99	
3	3	1_2	2024-05-11 14:25:45	1715437545	l3ng5xx	2024-05-12 01:04:01	1715475841	49	LadyGaga	0.99	
4	4	1_3	2024-05-11 14:25:45	1715437545	l3nhqi0	2024-05-12 01:16:14	1715476574	49	LadyGaga	0.99	
5	5	2	2024-05-11 14:25:45	1715437545	l3loj9b	2024-05-11 17:49:40	1715449780	49	LadyGaga	0.99	
6	6	2_1	2024-05-11 14:25:45	1715437545	l3m2ff6	2024-05-11 19:22:34	1715455354	49	LadyGaga	0.99	
7	7	2_1_1	2024-05-11 14:25:45	1715437545	l3n3nr2	2024-05-11 23:31:12	1715470272	49	LadyGaga	0.99	
8	8	2_1_1_1	2024-05-11 14:25:45	1715437545	l3nf1av	2024-05-12 00:55:27	1715475327	49	LadyGaga	0.99	
9	9	2_1_2	2024-05-11 14:25:45	1715437545	l3n8rvt	2024-05-12 00:08:13	1715472493	49	LadyGaga	0.99	
10	10	2_1_3	2024-05-11 14:25:45	1715437545	l3n8voe	2024-05-12 00:09:01	1715472541	49	LadyGaga	0.99	
11	11	2_2	2024-05-11 14:25:45	1715437545	l3nccvb	2024-05-12 00:35:10	1715474110	49	LadyGaga	0.99	
12	12	3	2024-05-11 14:25:45	1715437545	l3l3ckb	2024-05-11 15:29:58	1715441398	49	LadyGaga	0.99	
13	13	3_1	2024-05-11 14:25:45	1715437545	l3lmzsz	2024-05-11 17:39:19	1715449159	49	LadyGaga	0.99	
14	14	3_2	2024-05-11 14:25:45	1715437545	l3ltod7	2024-05-11 18:23:55	1715451835	49	LadyGaga	0.99	
15	15	3_2_1	2024-05-11 14:25:45	1715437545	l3n5oba	2024-05-11 23:45:28	1715471128	49	LadyGaga	0.99	
16	16	3_2_1_1	2024-05-11 14:25:45	1715437545	l3na9z2	2024-05-12 00:19:25	1715473165	49	LadyGaga	0.99	
17	17	3_3	2024-05-11 14:25:45	1715437545	l3l9dxq	2024-05-11 16:10:04	1715443804	49	LadyGaga	0.99	
18	18	3_4	2024-05-11 14:25:45	1715437545	l3ndpu1	2024-05-12 00:45:24	1715474724	49	LadyGaga	0.99	
19	19	4	2024-05-11 14:25:45	1715437545	l3l9421	2024-05-11 16:08:14	1715443694	49	LadyGaga	0.99	
20	20	5	2024-05-11 14:25:45	1715437545	l3lc348	2024-05-11 16:28:01	1715444881	49	LadyGaga	0.99	
21	21	6	2024-05-11 14:25:45	1715437545	l3l3i14	2024-05-11 15:30:59	1715441459	49	LadyGaga	0.99	

Figure 6 Result of Reddit thread data

2.2 Top 5 Actors

In this section, I will create actor networks from the collected data from the previous section and list the top 5 most influential actors for Lady Gaga's "Poker Face" and "Alejandro" YouTube videos, as well as for the Reddit thread discussing her meat dress. Note here, because I am planing to compare the result of “Poker Face" and "Alejandro", so I will create a separate network for this two videos. This analysis will help us understand the most active and influential users in the discussions surrounding these topics.

```
# Create actor network for Poke Face
yt_actor_network <- yt_data |> Create("actor")
yt_actor_graph <- yt_actor_network |> Graph()

# Remove the video node from the graph
video_node <- V(yt_actor_graph)[name == "VIDEOID:bESGLojNYSo"]
yt_actor_graph <- delete_vertices(yt_actor_graph, video_node)

# Calculate PageRank for Poke Face
rank_yt_actor <- sort(page_rank(yt_actor_graph)$vector, decreasing = TRUE)
top_yt_actors <- head(rank_yt_actor, 5)

# Replace IDs with display names
top_yt_actors_names <- sapply(names(top_yt_actors), function(x) {
  name <- yt_data$AuthorDisplayName[yt_data$AuthorChannelID == x]
  if (length(name) > 0) return(name[1]) else return(x)
})

# Create DataFrame for Poke Face
top_yt_actors_df <- data.frame(Author = top_yt_actors_names, PageRank = top_yt_actors)

# Plot top actors for Poke Face
ggplot(top_yt_actors_df, aes(x = reorder(Author, PageRank), y = PageRank, fill = Author)) +
  geom_col() +
  coord_flip() +
  labs(title = "Top PageRank Actors on YouTube Video: Poke Face",
       x = "Author",
       y = "PageRank Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
```

Figure 7 R code of analysing top 5 actor from Poker Face YouTube Video

```r
# Create actor network for Alejandro
alejandro_actor_network <- alejandro_yt_data |> Create("actor")
alejandro_actor_graph <- alejandro_actor_network |> Graph()

# Remove the video node from the graph
video_node_alejandro <- V(alejandro_actor_graph)[name == "VIDEOID:niqrrmev4mA"]
alejandro_actor_graph <- delete_vertices(alejandro_actor_graph, video_node_alejandro)

# Calculate PageRank for Alejandro
rank_alejandro_actor <- sort(page_rank(alejandro_actor_graph)$vector, decreasing = TRUE)
top_alejandro_actors <- head(rank_alejandro_actor, 5)

# Replace IDs with display names
top_alejandro_actors_names <- sapply(names(top_alejandro_actors), function(x) {
  name <- alejandro_yt_data$AuthorDisplayName[alejandro_yt_data$AuthorChannelID == x]
  if (length(name) > 0) return(name[1]) else return(x)
})

# Create DataFrame for Alejandro
top_alejandro_actors_df <- data.frame(Author = top_alejandro_actors_names, PageRank = top_alejandro_actors)

# Plot top actors for Alejandro
ggplot(top_alejandro_actors_df, aes(x = reorder(Author, PageRank), y = PageRank, fill = Author)) +
  geom_col() +
  coord_flip() +
  labs(title = "Top PageRank Actors on YouTube Video: Alejandro",
       x = "Author",
       y = "PageRank Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")

# Calculate how many unique actors there are
alejandro_unique_authors <- unique(alejandro_authors)
print(length(alejandro_unique_authors))
```

Figure 8 R code of analysing top 5 actor from Alejandro YouTube Video

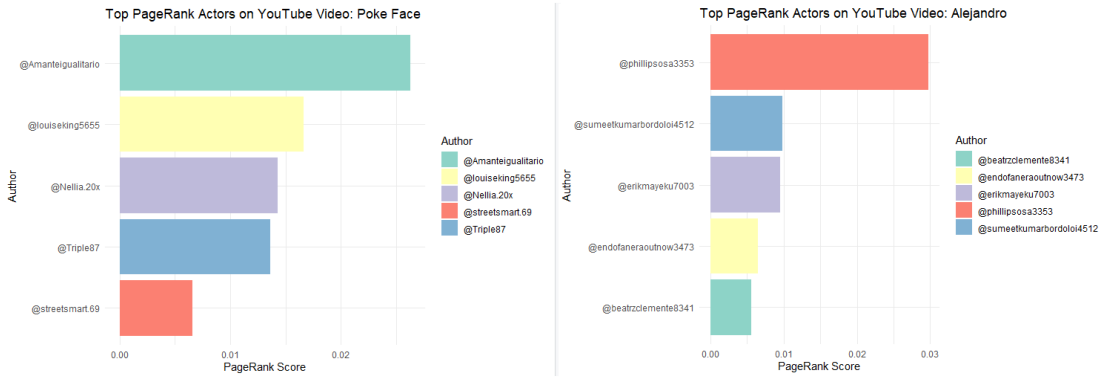


Figure 9 Top 5 actors on Lady Gaga's YouTube Videos: "Alejandro" and "Poker Face"

From the chart, it is clear that the top 5 commenters for Poker Face (on the left) have a relatively even number of score, around 0.015 each. On the right, the number of comments for Alejandro varies significantly, with the leading actor contributing up to 0.03 page rank score.

Referring back to the R code, the PageRank algorithm identifies the most influential commenters based on their connectivity within the network. In the context of YouTube

comments, this means users who interact with others frequently, either by replying to comments or being mentioned often.

Firstly, the PageRank analysis for the "Alejandro" video indicates that the most influential commenter is "@phillipsosa3353," who has the highest PageRank score among all users. This suggests that "@phillipsosa3353" frequently engages in discussions or receives a significant number of replies to their comments, making them a central figure in the conversation surrounding this video. Given the controversial nature of the "Alejandro" video, it is likely that "@phillipsosa3353" is actively debating their viewpoints, which could be attracting more interactions.

In contrast, the PageRank analysis for the "Poker Face" video shows a more balanced distribution of influence among the top users. The scores for these users are relatively close to each other, indicating a more evenly distributed network of influence. The "Poker Face" video has being one of Lady Gaga's most popular and mainstream songs, is more likely generates a broad range of interactions without the intense debate seen in the "Alejandro" video. This leads to a more diverse set of influential commenters, as no single user dominates the conversation.

```r
# Create Reddit actor network
reddit_actor_network <- rd_data |> Create("actor")
reddit_actor_graph <- reddit_actor_network |> Graph()

# Check available vertex attributes to find the correct one for usernames
print(vertex_attr_names(reddit_actor_graph))

# Assuming 'user' is the correct attribute, reassign it
V(reddit_actor_graph)$name <- V(reddit_actor_graph)$user

# Calculate PageRank
rank_reddit_actor <- sort(page_rank(reddit_actor_graph)$vector, decreasing = TRUE)

# Display top 5
top_reddit_page_rank <- head(rank_reddit_actor, 5)
print(top_reddit_page_rank)

# Convert to data frame for plotting
top_reddit_page_rank_df <- data.frame(
  Author = names(top_reddit_page_rank),
  PageRank_Score = as.numeric(top_reddit_page_rank),
  stringsAsFactors = FALSE
)

# Check the structure of the data frame
str(top_reddit_page_rank_df)

# Create the plot
ggplot(top_reddit_page_rank_df, aes(x = reorder(Author, PageRank_Score), y = PageRank_Score, fill = Author)) +
  geom_col() +
  coord_flip() +  # Flip the coordinates to make it horizontal; easier to read
  labs(title = "Top PageRank Actors on Reddit Thread: Lady Gaga Meat Dress",
       x = "Author",
       y = "PageRank Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")  # Adds a pleasant color palette
```

Figure 10 R code of analysing top 5 actor and unique actors from Reddit thread

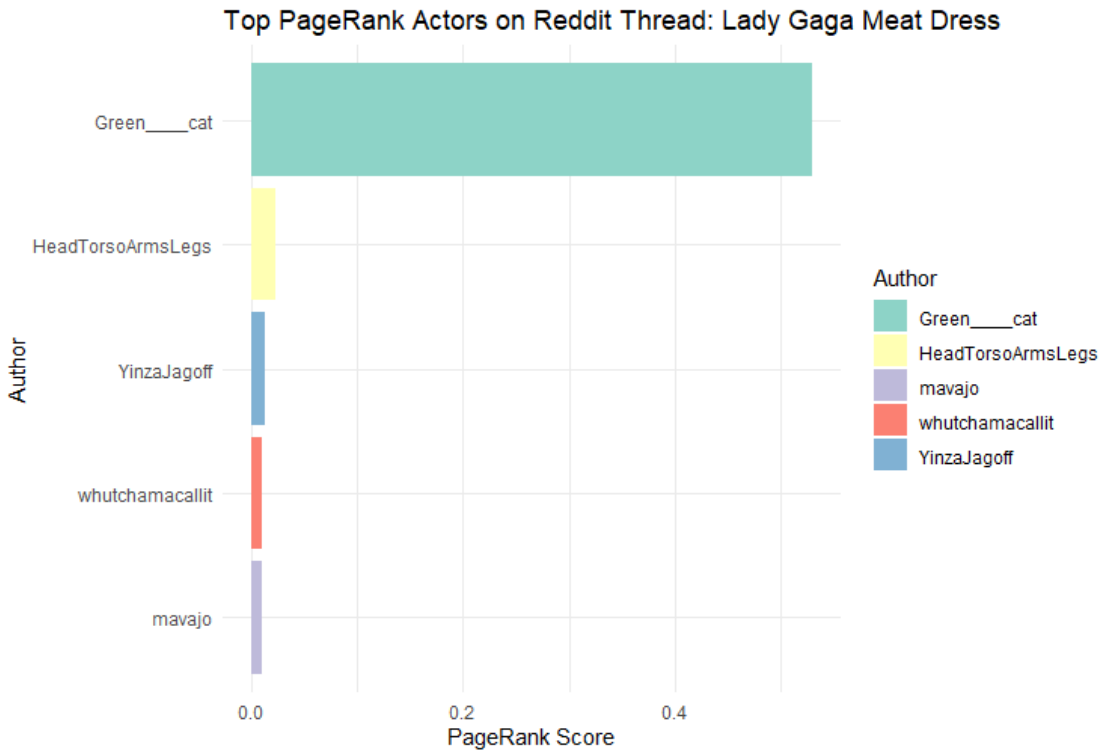


Figure 11 Top 5 Actors on Lady Gaga's Reddit thread

Compared to the YouTube videos, the PageRank score for the Reddit thread discussing Lady Gaga's meat dress shows a significant difference, with the user "Green____cat" having an exceptionally high influence. This typically indicates that "Green____cat" is at the center of the discussion, likely prompting a large number of replies or mentions, making them an unusually active actor. However, unlike YouTube, Reddit threads are initiated by a specific user, and most users will respond to this initiator, focusing the discussion around them. In this case, "Green____cat" is the originator of the thread, which explains why their score is exceptionally high.

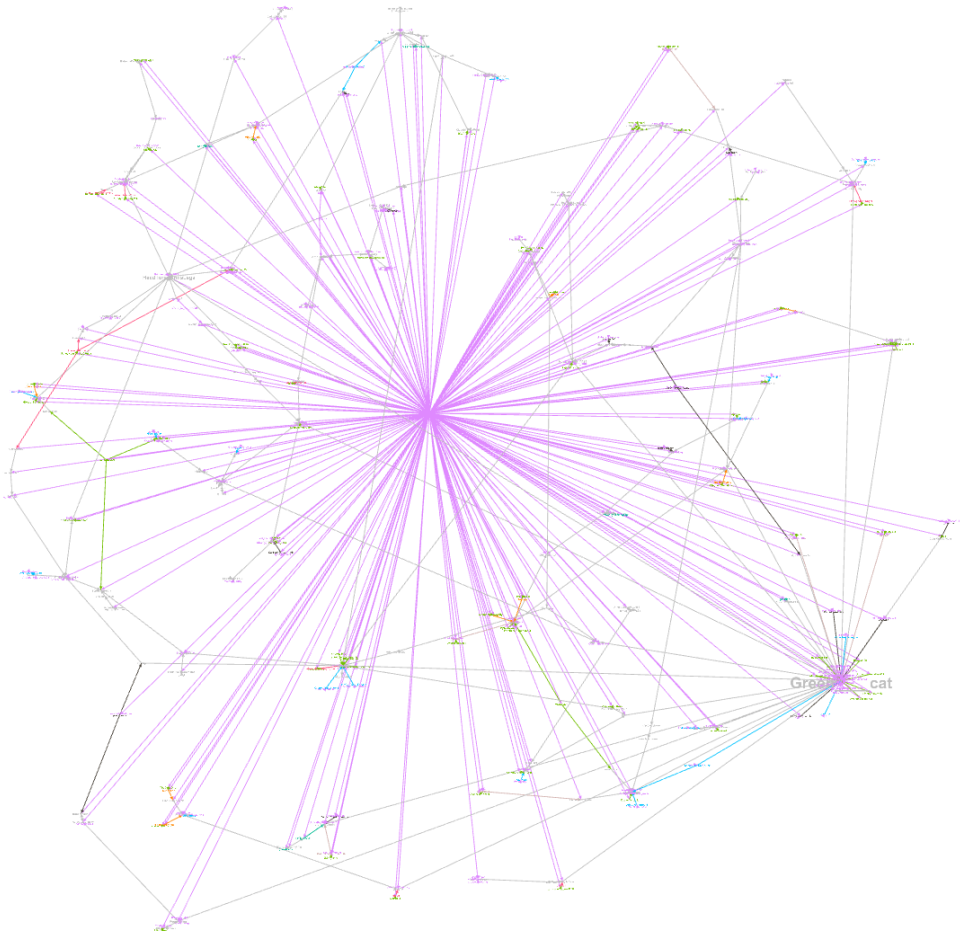


Figure 12 PageRank Graph of Alejandro from Gephi

The graph above generated using Gephi, represents the interaction network of a Reddit discussion thread about Lady Gaga's meat dress incident. Each node in the graph represents a user, and the size of the nodes typically indicates the importance or activity level of the user within the network. The edges between the nodes represent interactions between users, and the color and thickness of these edges may indicate the strength or frequency of these interactions.

The most prominent central node is "Green____cat," which has many direct connections to other nodes. This suggests that "Green____cat" is the initiator or a major participant in the discussion, attracting a lot of replies and interactions.

Other users interact less frequently and mainly with the central node, indicating the discussion is centered around a single user. The graph provides a visual representation of the discussion's concentration and the influence of the main participants. This result is consistent with findings generated using R coding, particularly the exceptionally high PageRank score for "Green____cat," confirming their significant influence in the discussion.

2.3 Unique Actors

```
# Calculate how many unique actors in Poke Face
poke_face_authors <- yt_data$AuthorDisplayName
poke_face_unique_authors <- unique(poke_face_authors)
print(length(poke_face_unique_authors))

#------------------------------------------------------------------------

# Calculate how many unique actors in Alejandro
alejandro_authors <- alejandro_yt_data$AuthorDisplayName
alejandro_unique_authors <- unique(alejandro_authors)
print(length(alejandro_unique_authors))

#------------------------------------------------------------------------

# Calculate how many unique actors there are in the Reddit thread
reddit_authors <- rd_data$user
reddit_unique_authors <- unique(reddit_authors)
print(length(reddit_unique_authors))
```

Figure 13 R code of unique actors

The above figure shows the R code for unique actors from both youtube and Reddit data. In the case of alejandro, the 'unique()' function with the parameter "alejandro_authors" is used to generate the number of unique authors, which is then stored in the "alejandro_unique_authors". The parameter 'alejandro_authors' stores the names of the authors of "Alejandro". The third line is a 'print()' plus 'length()' function, which prints out the number of 'alejandro_unique_authors' and thus gives the number of Unique actors.

Topic	Total number of unique actors	Total number of Comments
Poke Facce	3036	3652
Alejandro	3050	3574
Meat Dress	436	720

Figure 14 Number of unique actors from both youtube videos and Reddit thread.

The youtube videos have significantly higher engagement levels compared to the Reddit thread about the meat dress, both in terms of the number of unique actors and the total number of comments. This indicates that the YouTube platform garners more widespread engagement for Lady Gaga's content compared to the Reddit platform for that particular topic.

Alejandro has the highest number of unique actors (3050), slightly more than Poker Face (3036), while the Poker Face received the highest number of total comments (3652), followed closely by Alejandro (3574). The Meat Dress thread has only 720

comments, indicating lower overall activity compared to the YouTube videos.

if we make a ratio between unique actors and comments, we will notice YouTube videos have very close ratio around 0.8, but Reddit only has 0.6.

$$\text{Alejandro: } \frac{3050}{3574} = 0.85$$

$$\text{Poker Face: } \frac{3036}{3652} = 0.83$$

$$\text{Reddit thread: } \frac{436}{720} = 0.6$$

The data demonstrate Lady Gaga's strong presence and reach on YouTube, with high levels of engagement from a large number of unique viewers. The Reddit thread on the meat dress, despite having fewer participants, shows that certain topics can generate intense discussion among a smaller group of users.

2.4 Spotify Data Retrieval

```
# Set up authentication variables
app_id <- "4a9a2950b79b4adfbd182d5ff47c359d"
app_secret <- "01075fcec4944fa3873e17d3f4a74bca"
token <- "1"

# Authentication for spotifyr package:
Sys.setenv(SPOTIFY_CLIENT_ID = app_id)
Sys.setenv(SPOTIFY_CLIENT_SECRET = app_secret)
access_token <- get_spotify_access_token()


# Search for the artist 'Lady Gaga' and get her Spotify ID
artist_name <- "Lady Gaga"
search_results <- search_spotify(artist_name, type = "artist")
lady_gaga_info <- search_results[search_results$name == artist_name, ]
View(search_results)
View(lady_gaga_info)
```

Figure 15 R code Spotify Search for Lady Gaga

The above code is used to authenticate with the Spotify API and retrieve data about the artist "Lady Gaga." These variables app_id and app_secret store the Spotify application credentials.

	genres	href	id	images	name
1	c("art pop", "dance pop", "pop")	https://api.spotify.com/v1/artists/1HY2Jd0NmPuamShAr6K...	1HY2Jd0NmPuamShAr6KMms	*2 variables*	Lady Gaga
2	character(0)	https://api.spotify.com/v1/artists/0xukYGKRRwBWS1N9sfCQ...	0xukYGKRRwBWS1N9sfCQoq	*2 variables*	Gizzle
3	character(0)	https://api.spotify.com/v1/artists/5YWaklg9jWNObRq8369j9y	5YWaklg9jWNObRq8369j9y	*2 variables*	Lady Gaga FKA
4	character(0)	https://api.spotify.com/v1/artists/6dvIEb67jvtYQnnXrgwrxJ	6dvIEb67jvtYQnnXrgwrxJ	*list()*	Lady G
5	character(0)	https://api.spotify.com/v1/artists/08VVNBMMHgFDNuWUA...	08VVNBMMHgFDNuWUA3ez6O	*2 variables*	Lady Gaga Mindfulness
6	c("alternative dance", "metropopolis", "neo-synthp [...]	https://api.spotify.com/v1/artists/5TfnQ0Ai1cEbKY5katFK14	5TfnQ0Ai1cEbKY5katFK14	*2 variables*	Ladyhawke
7	character(0)	https://api.spotify.com/v1/artists/7BXMFyPdBOP61qeL3E5q...	7BXMFyPdBOP61qeL3E5qU7	*2 variables*	Gaga For Lady Stars
8	character(0)	https://api.spotify.com/v1/artists/5tDlgMd5IZh4DFbR9NaMfi	5tDlgMd5IZh4DFbR9NaMfi	*2 variables*	Lady Harmony
9	character(0)	https://api.spotify.com/v1/artists/02OqjFXKgtAtRpr8CRaZjv	02OqjFXKgtAtRpr8CRaZjv	*list()*	Lady Gaga's Karaoke Band
10	c("dancehall queen", "old school dancehall")	https://api.spotify.com/v1/artists/1C5RzkueJX7bh2ZrzpA2fa	1C5RzkueJX7bh2ZrzpA2fa	*2 variables*	Lady G
11	character(0)	https://api.spotify.com/v1/artists/78RN4N0o7KIcN63nH8Twxa	78RN4N0o7KIcN63nH8Twxa	*2 variables*	Lady Gaga Orkestband
12	c("afrobeats", "afropop", "igbo pop", "nigerian po [...]	https://api.spotify.com/v1/artists/62s0EsXQNJEwy8fKZ386VU	62s0EsXQNJEwy8fKZ386VU	*2 variables*	Larry Gaaga
13	character(0)	https://api.spotify.com/v1/artists/7oaK0rVl323hWZPoxx5ZVT	7oaK0rVl323hWZPoxx5ZVT	*2 variables*	Lady Gaga - Piano Covers
14	rumba catalana	https://api.spotify.com/v1/artists/111HTVEie3XPN4ooluaVXq	111HTVEie3XPN4ooluaVXq	*list()*	Lady Gipsy
15	fake	https://api.spotify.com/v1/artists/38HjA3mCsjOcAgJiJLsx2D	38HjA3mCsjOcAgJiJLsx2D	*2 variables*	Lady Gaga for Piano
16	character(0)	https://api.spotify.com/v1/artists/6j2gN96Z5ZJYnZsIRqoqWr	6j2gN96Z5ZJYnZsIRqoqWr	*2 variables*	Lady Gee
17	character(0)	https://api.spotify.com/v1/artists/47l1lo01l1zubnuzDPb9OB	47l1lo01l1zubnuzDPb9OB	*2 variables*	Lady Gaga Tribute Orchestra
18	character(0)	https://api.spotify.com/v1/artists/2wfjcgwppMkXwMuYwvN...	2wfjcgwppMkXwMuYwvN5mx	*list()*	Lady Gatica
19	character(0)	https://api.spotify.com/v1/artists/26v8sW8b04x0LmkPIiYr4d	26v8sW8b04x0LmkPIiYr4d	*2 variables*	Tribute To Lady Gaga
20	dark techno	https://api.spotify.com/v1/artists/0kolce4lQQT3kf9QJmEuMX	0kolce4lQQT3kf9QJmEuMX	*2 variables*	Gaga

Figure 16 Lady Gaga Spotify search results

How many years has she been active

To find out how long Lady Gaga has been active, we can find out the release date of her earliest album.

```r
# Fetch albums and extract the earliest release date
albums <- get_artist_albums(lady_gaga_info$id, include_groups = "album", limit = 50)
album_dates <- as.Date(albums$release_date)
View(album_dates)
# Calculate how many years they have been active
years_active <- as.numeric(difftime(Sys.Date(), min(album_dates), units = "days")) / 365.25
# Print the years active
print(paste("Lady Gaga has been active for", years_active, "years."))
```

Figure 17 R code for active years in Spotify

Here, we get Lady Gaga's album data from Spotify and collect the release dates. Convert these dates into a usable date object format. Determine the earliest release date from which to start her music career. Then calculate the total duration in days from the earliest release date to the current date. Finally divide this by 365 days to get the answer in years.

```
[1] "Lady Gaga has been active for 16.3860369609856 years."
```

Figure 18 result of active years

It is worth noting that this coincides with the active years mentioned in the introduction. This justifies that the methodology of finding her active times is correct and accurate.

How many albums and songs has she published

```r
# Retrieve percific album data of artist
albums <- get_artist_albums("1HY2JdONmPuamShAr6KMms",
                            include_groups = c("album", "single", "appears_on", "compilation"))
View(albums)

unique_albums <- unique(albums$name)
num_albums <- length(unique_albums)

# Fetch all tracks for each album
all_tracks <- list()
for (album_id in albums$id) {
  tracks <- get_album_tracks(album_id)
  all_tracks <- c(all_tracks, tracks$name)
}
unique_tracks <- unique(all_tracks)
num_tracks <- length(unique_tracks)

# Print the number of albums and tracks
print(paste("Lady Gaga has published", num_albums, "albums."))
print(paste("Lady Gaga has published", num_tracks, "songs."))
```

Figure 19 R code for number of songs and albums

in here, I fetched all types of albums (including singles and compilations), to get the total number of songs. And then retrieves and counts the unique albums and tracks.

```
> print(paste("Lady Gaga has published", num_albums, "albums."))
[1] "Lady Gaga has published 20 albums."
> print(paste("Lady Gaga has published", num_tracks, "songs."))
[1] "Lady Gaga has published 181 songs."
```

Figure 20 Result of number of songs and albums

It is worth noting that, referring to the introduction, the number of albums and songs retrieved from Spotify data is much higher than mentioned in the introduction. A key reason is that Spotify not only counts the original versions of the songs but also includes remixed versions or mixed versions. Sometimes, when a song is performed in collaboration with different artists, it is counted as different songs. Additionally, Spotify might have some unclear data, which could also increase the count of albums and songs.

Collaborated artist:

```
# Retrieve information about Lady Gaga' related artists

related_bm <- get_related_artists("1HY2JdONmPuamShAr6KMms")
View(related_bm)


# Create a network of artists related to Lady Gaga

edges <- c()
for (artist in related_bm$id){
  related <- get_related_artists(artist)
  artist_name <- get_artist(artist)$name
  for (relatedartist in related$name){
    edges <- append(edges, artist_name)
    edges <- append(edges, relatedartist)
  }
}
edges[1:10]
```

Figure 21 R code for collaborated artist

Here, we call the get_lated_artists function to retrieve a list of artists related to Lady Gaga. We then loop through the IDs of each artist we find, get their names, and then display the top 10 artists.

```
> edges[1:10]
 [1] "Britney Spears"     "Spice Girls"        "Britney Spears"     "Christina Aguilera" "Britney Spears"
 [6] "Jennifer Lopez"     "Britney Spears"     "Gwen Stefani"       "Britney Spears"     "The Pussycat Dolls"
```

Figure 22 Result of collaborated artist

The artists listed are iconic figures in pop music, indicating that Lady Gaga's related artists network is deeply embedded in the pop genre. The connections include both groups (Spice Girls, The Pussycat Dolls) and solo artists (Christina Aguilera, Gwen Stefani, Jennifer Lopez). This diversity shows that Lady Gaga's influence and connections span across different formats of music groups.

Prevalent Features

```r
#get the flame albums
songs <- get_album_tracks("1jpUMnKpRlng1OJN7LJauV")
View(songs)


# Retrieve song data
song <- get_track_audio_features("5R8dQOPq8haW94K7mgERlO")
View(song)

# Get audio features for it
audio_features <- get_artist_audio_features(lady_gaga_id) # artist ID for Lady Gaga
View(audio_features)

audio_features <- audio_features[!duplicated(audio_features$track_name), ]


# Plot valence scores for each album
ggplot(audio_features, aes(x = valence, y = album_name)) +
  geom_density_ridges() +
  theme_ridges() +
  ggtitle("Happiness in Lady Gaga Albums")

# Correct title and plot for energy scores
ggplot(audio_features, aes(x = energy, y = album_name)) +
  geom_density_ridges() +
  theme_ridges() +
  ggtitle("Energy in Lady Gaga Albums")
```

Figure 23 R code for 2 prevalent features in Spotify

Here, I use the album ID to retrieve tracks from a specific album, and then use the track ID to retrieve detailed audio features for a specific track, thus retrieving audio features for all of Lady Gaga's tracks. This is followed by eliminating duplicate entries in the dataset. Finally, visualize the distribution of Happiness Index and Energy Score for each album.

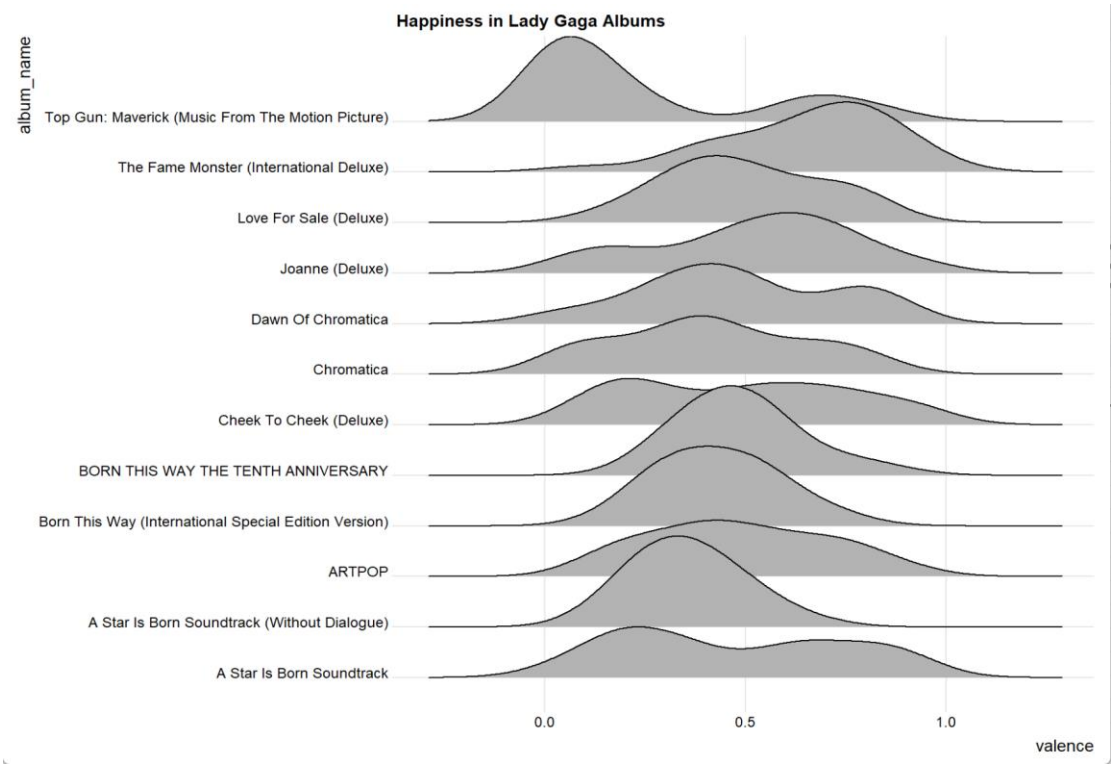


Figure 24 Valence plot of Lady Gaga's track

Valence is a measure of the musical positivity conveyed by a track, with higher scores indicating happier and more uplifting music. This valence plot visualizes the distribution of valence (happiness) scores across various Lady Gaga albums.

Top Gun: Maverick:
The tracks from this album show a wide range of valence scores, with a noticeable peak around the low range around 0.1. This suggests that many songs on this album are perceived as having a more somber or neutral emotional tone.

The Fame Monster:
This album has a higher concentration of tracks with moderate to high valence scores about 0.7. This indicates that many songs on this album are perceived as positive and uplifting.

Cheek to cheek & A Star Is Born Soundtrack:
The distribution of these two tracks is relatively even, with both having 2 different peaks around 0.3 and 0.6, which suggests that the songs in these 2 tracks are both very downbeat and positive, rather than a single style of music.

The peak range of all the remaining tracks is centred around 0.4 and 0.5, which shows that the main themes of Lady Gaga's songs are still quite diverse, ranging from joyful and lyrical to muffled and sad.

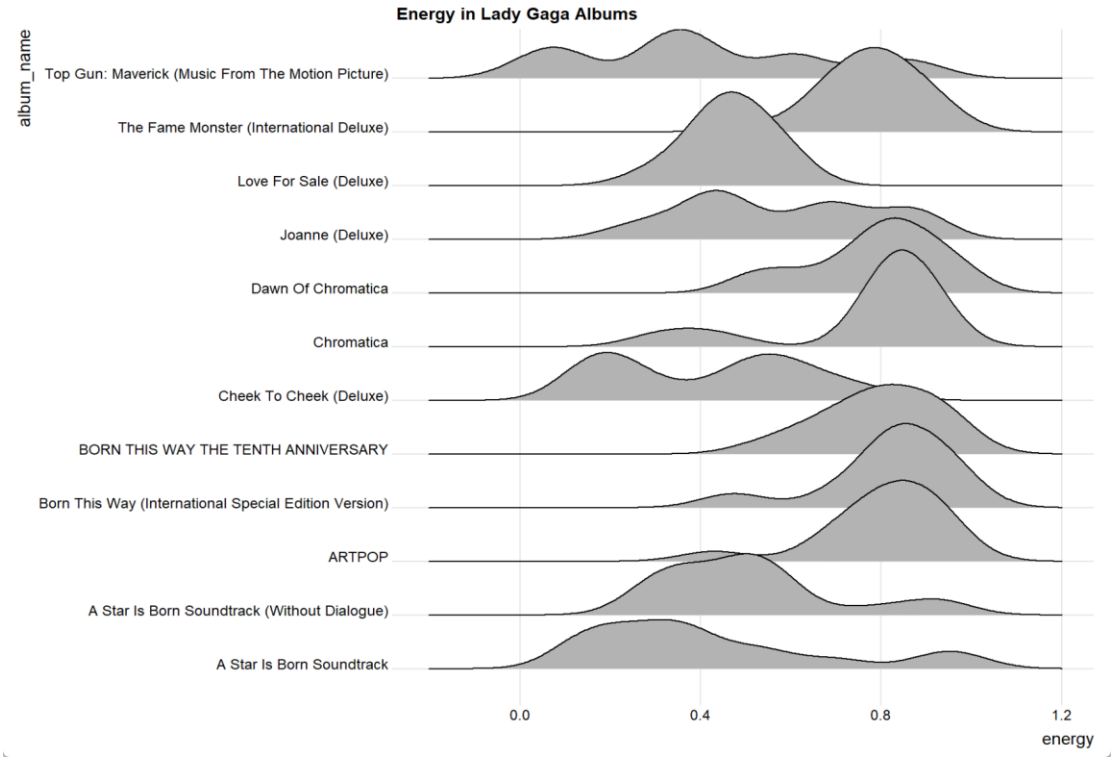


Figure 25 Energy plot of Lady Gaga's track

Many of Lady Gaga's albums, particularly "The Fame Monster," "Chromatica", "Born this way the tenth anniversary", and "ARTPOP," have a high concentration of tracks with high energy scores, reflecting her dynamic and upbeat musical style.

Top Gun: Maverick & Check to Check & Joanne:
The tracks from these albums show a wide range of energy scores, with most of the albums featuring either very high or very low energy levels. Only these three albums have a less compact distribution, indicating that the songs do not exhibit a clear energy tendency and are relatively moderate. The average energy score for these three albums is below 0.5, suggesting that these songs are generally less active.

Love for Sale & A Start is Bom Soundtrack:
"Love for Sale" & "A Star Is Born Soundtrack": The overall energy in these two albums is lower, with peaks between 0.4 and 0.5. These albums have a softer and more subdued feel compared to her high-energy albums. This is particularly evident in "A Star Is Born Soundtrack," which features more ballads and emotional songs.

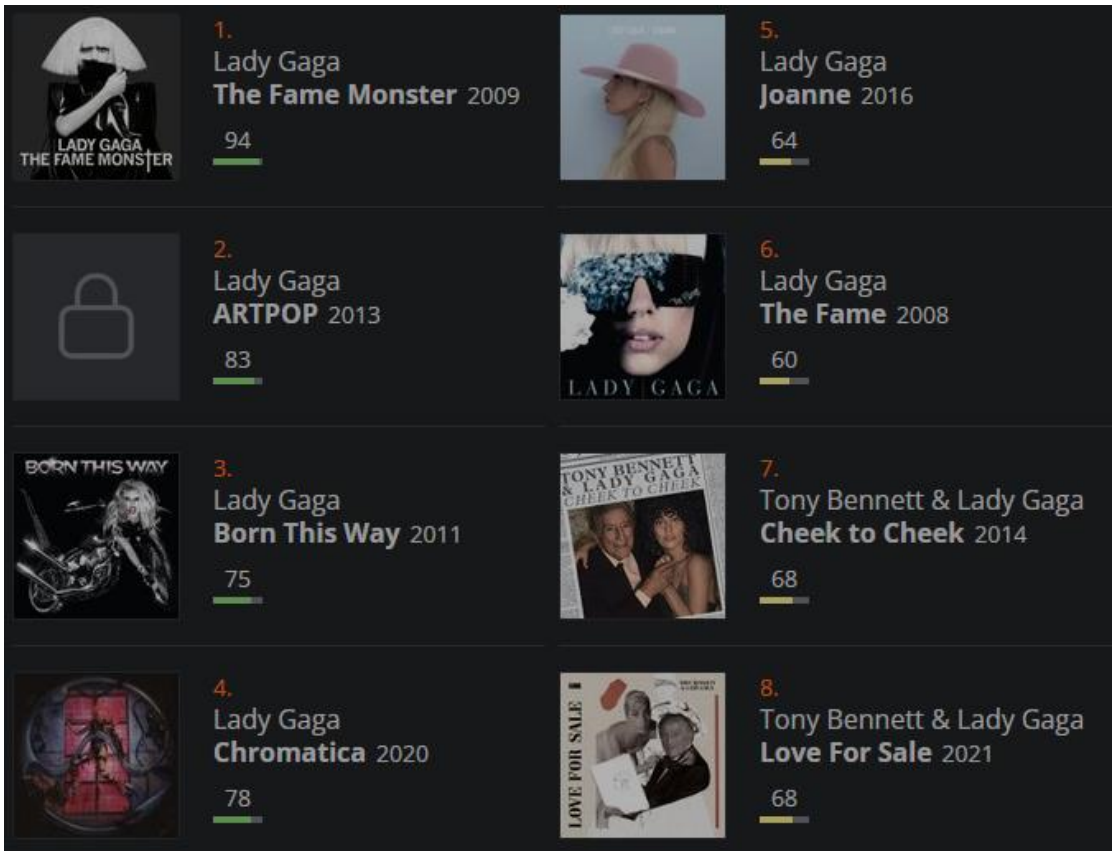


Figure 26 Lady Gaga Alums Ranked

The image above is from a poll of ladygaga fans called AOTY, who voted on their favourite of the eight albums (Godga, 2023).

When comparing the energy and happiness (valence) scores of Lady Gaga's albums with their rankings, we observe a significant correlation between these musical features and the albums' popularity.

The top 3 rank albums “The Fame Monster”, ”ARTPOP” and “Born This Way” feature many fast-paced, loud, and vibrant tracks, aligning with Lady Gaga's dynamic musical style. This high energy level likely contributes to their popularity, as energetic songs are often more engaging and suitable for dance and party settings. Additionally, the relatively high valence scores suggest that these albums also include positive and upbeat tracks. Songs that evoke positive emotions can be more appealing to a wider audience, contributing to the albums' success.

“Cheek to Cheek” and “Love For Sale”, these two albums show lower energy and happiness scores, which feature jazz standards and collaborations with Tony Bennett, have significantly lower energy levels. These albums are more mellow and relaxed, which may appeal to a niche audience but not to the broader pop market.

3. Text Pre-Processing

```r
# Prepare text data for Poke Face and perform text pre-processing
comments <- yt_data$Comment

clean_text <- comments |>
  tolower() |>
  replace_url() |>
  replace_html() |>
  replace_non_ascii() |>
  replace_word_elongation() |>
  replace_internet_slang() |>
  replace_contraction() |>
  removeNumbers() |>
  removePunctuation() |>
  replace_emoji() |>
  replace_emoticon()

# Create a corpus
text_corpus <- VCorpus(VectorSource(clean_text))

# Further text pre-processing
text_corpus <- text_corpus |>
  tm_map(content_transformer(tolower)) |>
  tm_map(removeWords, stopwords(kind = "SMART")) |>
  tm_map(stripWhitespace)
```

```r
# Prepare text data for Alejandro and perform text pre-processing
alejandro_comments <- alejandro_yt_data$Comment

clean_text_alejandro <- alejandro_comments |>
  tolower() |>
  replace_url() |>
  replace_html() |>
  replace_non_ascii() |>
  replace_word_elongation() |>
  replace_internet_slang() |>
  replace_contraction() |>
  removeNumbers() |>
  removePunctuation() |>
  replace_emoji() |>
  replace_emoticon()

# Create a corpus
text_corpus_alejandro <- VCorpus(VectorSource(clean_text_alejandro))

# Further text pre-processing
text_corpus_alejandro <- text_corpus_alejandro |>
  tm_map(content_transformer(tolower)) |>
  tm_map(removeWords, stopwords(kind = "SMART")) |>
  tm_map(stripWhitespace)
```

Figure 27 Poker Face (left) and Alejandro (right) text pre-processing

```r
# Prepare text data for Reddit and perform text pre-processing
reddit_comments <- rd_data$comment

clean_text_reddit <- reddit_comments |>
  tolower() |>
  replace_url() |>
  replace_html() |>
  replace_non_ascii() |>
  replace_word_elongation() |>
  replace_internet_slang() |>
  replace_contraction() |>
  removeNumbers() |>
  removePunctuation() |>
  replace_emoji() |>
  replace_emoticon()

# Create a corpus
text_corpus_reddit <- VCorpus(VectorSource(clean_text_reddit))

# Further text pre-processing
text_corpus_reddit <- text_corpus_reddit |>
  tm_map(content_transformer(tolower)) |>
  tm_map(removeWords, stopwords(kind = "SMART")) |>
  tm_map(stripWhitespace)
```

Figure 28 Reddit thread text pre-processing

As graphs show above, a text pre-processing is done by data cleaning. Here, we remove some words in the data that prevent data analysis, such as URL, emji, html code, etc. Next, we delete some 'stop words' that are not commonly used and are not helpful for data analysis. Finally, we get a series of pure data which is helpful for analysis.

3.1 Term-Document Matrices

```r
# Create Document-Term Matrix
doc_term_matrix_reddit <- DocumentTermMatrix(text_corpus_reddit)

# Convert to data frame
dtm_df_reddit <- as.data.frame(as.matrix(doc_term_matrix_reddit))

# Calculate term frequencies
freq_reddit <- sort(colSums(dtm_df_reddit), decreasing = TRUE)

# Get top 10 most frequent terms
top_terms_reddit <- head(freq_reddit, 10)

# Plot the top 10 most frequent words
word_frequ_df_reddit <- data.frame(word = names(top_terms_reddit), freq = top_terms_reddit)
ggplot(word_frequ_df_reddit, aes(x = reorder(word, freq), y = freq, fill = word)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 Most Frequent Words in Reddit Comments",
       x = "Words",
       y = "Frequency") +
  theme_minimal()
```

Figure 29 R code for Reddit thread TDM

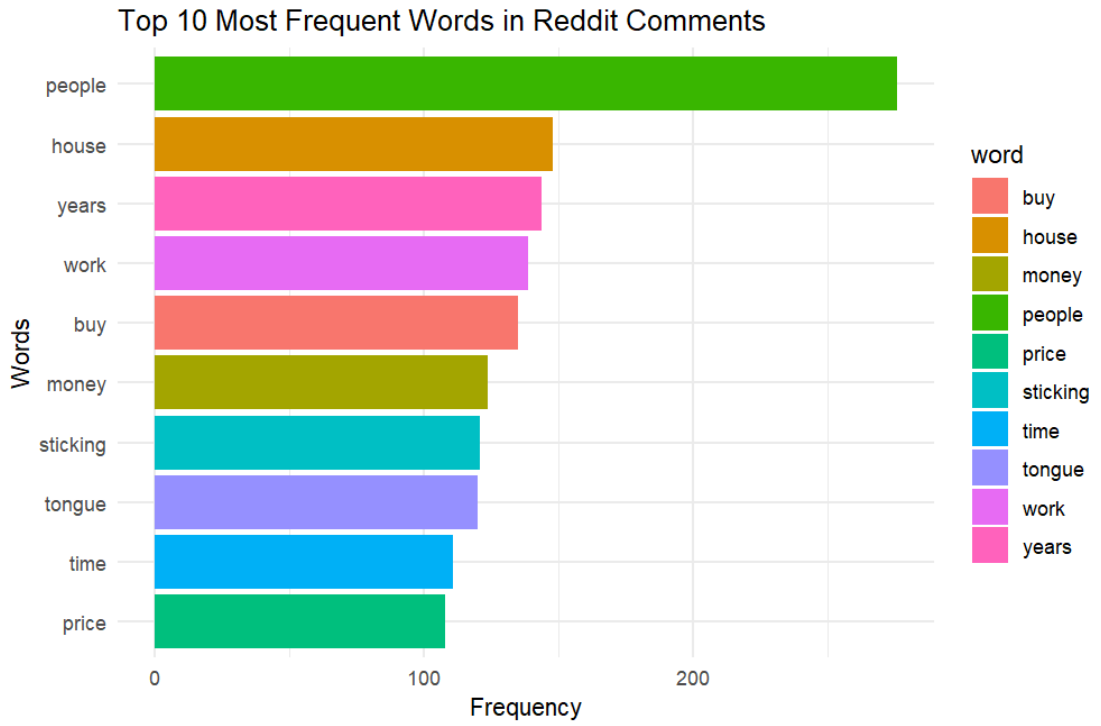


Figure 30 Top 10 most frequent word on Reddit Thread: "Meat Address"

The 10 most Frequent words in Reddit comments provide insight into the main themes discussed in the thread, so let's break them down one by one.

people:

This word appears the most frequently, indicating that discussions often involve

references to "people" and revolve around general societal issues, behaviors, or experiences. This could be due to commenters talking about the general public's reaction to Lady Gaga's meat dress or opinions expressed by various individuals.

house:
Frequent references to ‘house’ indicate that there is a lot of discussion related to housing. This could be a result of the meat skirt issue, which triggered an exploration of societal issues that extended to money, housing, etc.

years:
The term "years" is often used to refer to timeframes, indicating that commenters are discussing events or trends over the years. As the “meat dress” incident occurred in 2010, one might repeatedly refer to the timeline to emphasise that this is not content that has occurred in recent years

work:
The word "work" is common, highlighting discussions about employment, job conditions, work experiences, or labor market trends. It reflects the importance of job-related issues within the topic.

buy:
The word ‘buy’ is very much related to the word ‘dress’, and the commenter is most likely criticizing or discussing Lady Gaga's purchase of the dress.

money:
The term ‘money’ is popular and refers to discussions about finances, the state of the economy and financial decisions. Commenters may talk about their personal or Lady Gaga's financial situation. The word is also strongly associated with ‘buy’ and ‘dress’, and in all probability they will appear as a phrase, which explains why they are ranked so close together.

sticking:
This word could be part of specific phrases or metaphors used in the comments. It may relate to how the dress was perceived or remembered by people.

tongue:
This term might be used metaphorically or literally in the context of the meat dress, as it is an unusual and provocative piece of clothing.

time:
With reference to ‘year’, frequent references to ‘time’ may refer to the timing of events, the duration of impacts, or reflections on how ideas change over time.

price:
With reference to ‘money’ and ‘buy’, the word "price" appears often, highlighting discussions about the cost of goods, price changes, and affordability. This indicates that economic conditions and the cost of living are significant concerns for the commenters.

The frequent appearance of these 10 words indicates a focus on discussing the meat dress topic. The topic itself has high discussion potential and easily extends to societal issues. The words reflect the commenters' interests in general societal reactions, economic conditions, and how the event has been perceived over time.

```
# Create Document-Term Matrix
doc_term_matrix <- DocumentTermMatrix(text_corpus)

# Convert to data frame
dtm_df <- as.data.frame(as.matrix(doc_term_matrix))

# Calculate term frequencies
freq <- sort(colSums(dtm_df), decreasing = TRUE)

# Get top 10 most frequent terms
top_terms <- head(freq, 10)

# Plot the top 10 most frequent words
word_frequ_df <- data.frame(word = names(top_terms), freq = top_terms)
ggplot(word_frequ_df, aes(x = reorder(word, freq), y = freq, fill = word)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 Most Frequent Words in Poke Face Comments",
       x = "Words",
       y = "Frequency") +
  theme_minimal()
```

Figure 31 R code for Poker Face TDM

```
# Create Document-Term Matrix
doc_term_matrix_alejandro <- DocumentTermMatrix(text_corpus_alejandro)

# Convert to data frame
dtm_df_alejandro <- as.data.frame(as.matrix(doc_term_matrix_alejandro))

# Calculate term frequencies
freq_alejandro <- sort(colSums(dtm_df_alejandro), decreasing = TRUE)

# Get top 10 most frequent terms
top_terms_alejandro <- head(freq_alejandro, 10)

# Plot the top 10 most frequent words
word_frequ_df_alejandro <- data.frame(word = names(top_terms_alejandro), freq = top_terms_alejandro)
ggplot(word_frequ_df_alejandro, aes(x = reorder(word, freq), y = freq, fill = word)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 Most Frequent Words in Alejandro Comments",
       x = "Words",
       y = "Frequency") +
  theme_minimal()
```

Figure 32 R code for Alejandro TDM

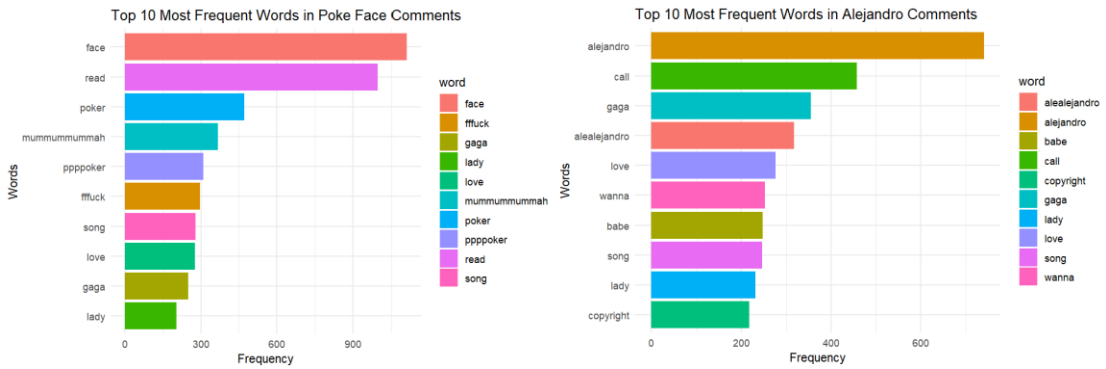


Figure 33 Top 10 most frequent word on YouTube Videos: "Alejandro" and "Poker Face"

Now let's analyse the song reviews from youtube. Due to the nature of music song comments, song titles, lyrics, and the artist's name frequently appear. For example, in the song 'Poker Face,' terms like 'face,' 'poker,' 'mumummummmah,' and 'pppppoker' are common, while in the song 'Alejandro,' terms like 'Alejandro,' 'call,' 'alealejandro,' 'wanna,' and the artist's name 'Lady' and 'Gaga' are prevalent. The frequent appearance of these words demonstrates how specific wording in the songs resonates with listeners, or they might be quoting these terms to describe their feelings or relate to the song's narrative. This frequent appearance of content usually signifies that the comments are positive and engaging. But there are also several controversial terms in it:

fffuck:
One of the words in the lyrics, but the presence of the word may also indicate an emotional or intense reaction by the reviewer in a positive or negative context. It may also be a misinterpretation or playful adaptation of the lyrics.

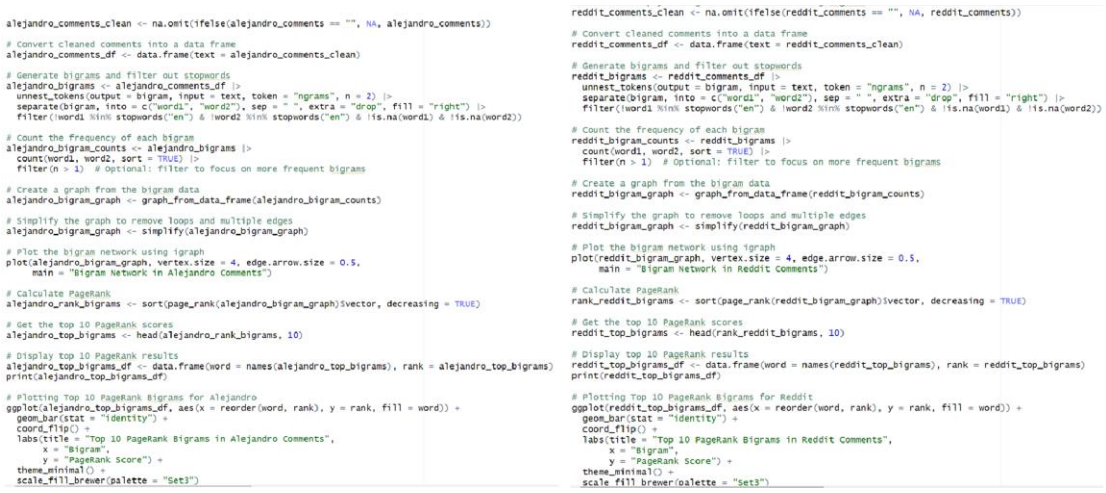
copyright:
This term may indicate a discussion of the copyright of the song or music video. It might be about concerns that the lyrics or the music video contain elements that are not in accordance with copyright laws. The word itself is neutral, but in the context of a song video, it tends to have a more negative connotation, as it often brings up legal or ethical issues.

The frequent words appearance of these words demonstrates how specific wording in the songs resonates with listeners, or they might be quoting these terms to describe their feelings or relate to the song's narrative. This frequent appearance of content usually signifies that the comments are positive and engaging. Additionally, terms like "copyright" suggest discussions about legal aspects, which tend to have a more negative connotation when associated with music videos.

3.2 Semantic Network

```r
comments_clean <- na.omit(ifelse(comments == "", NA, comments))

# Convert cleaned comments into a data frame
comments_df <- data.frame(text = comments_clean)

# Generate bigrams and filter out stopwords
bigrams <- comments_df |>
  unnest_tokens(output = bigram, input = text, token = "ngrams", n = 2) |>
  separate(bigram, into = c("word1", "word2"), sep = " ", extra = "drop", fill = "right") |>
  filter(!word1 %in% stopwords("en") & !word2 %in% stopwords("en") & !is.na(word1) & !is.na(word2))

# Count the frequency of each bigram
bigram_counts <- bigrams |>
  count(word1, word2, sort = TRUE) |>
  filter(n > 1)  # Optional: filter to focus on more frequent bigrams

# Create a graph from the bigram data
bigram_graph <- graph_from_data_frame(bigram_counts)

# Simplify the graph to remove loops and multiple edges
bigram_graph <- simplify(bigram_graph)

# Plot the bigram network using igraph
plot(bigram_graph, vertex.size = 4, edge.arrow.size = 0.5,
     main = "Bigram Network in Poke Face Comments")

# Calculate PageRank
rank_bigrams <- sort(page_rank(bigram_graph)$vector, decreasing = TRUE)

# Get the top 10 PageRank scores
top_bigrams <- head(rank_bigrams, 10)

# Display top 10 PageRank results
top_bigrams_df <- data.frame(word = names(top_bigrams), rank = top_bigrams)
print(top_bigrams_df)

# Plotting Top 10 PageRank Bigrams for Poke Face
ggplot(top_bigrams_df, aes(x = reorder(word, rank), y = rank, fill = word)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 PageRank Bigrams in Poke Face Comments",
       x = "Bigram",
       y = "PageRank Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
```

Figure 34 R code for Poker Face Semantic Network



```r
alejandro_comments_clean <- na.omit(ifelse(alejandro_comments == "", NA, alejandro_comments))

# Convert cleaned comments into a data frame
alejandro_comments_df <- data.frame(text = alejandro_comments_clean)

# Generate bigrams and filter out stopwords
alejandro_bigrams <- alejandro_comments_df |>
  unnest_tokens(output = bigram, input = text, token = "ngrams", n = 2) |>
  separate(bigram, into = c("word1", "word2"), sep = " ", extra = "drop", fill = "right") |>
  filter(!word1 %in% stopwords("en") & !word2 %in% stopwords("en") & !is.na(word1) & !is.na(word2))

# Count the frequency of each bigram
alejandro_bigram_counts <- alejandro_bigrams |>
  count(word1, word2, sort = TRUE) |>
  filter(n > 1)  # Optional: filter to focus on more frequent bigrams

# Create a graph from the bigram data
alejandro_bigram_graph <- graph_from_data_frame(alejandro_bigram_counts)

# Simplify the graph to remove loops and multiple edges
alejandro_bigram_graph <- simplify(alejandro_bigram_graph)

# Plot the bigram network using igraph
plot(alejandro_bigram_graph, vertex.size = 4, edge.arrow.size = 0.5,
     main = "Bigram Network in Alejandro Comments")

# Calculate PageRank
alejandro_rank_bigrams <- sort(page_rank(alejandro_bigram_graph)$vector, decreasing = TRUE)

# Get the top 10 PageRank scores
alejandro_top_bigrams <- head(alejandro_rank_bigrams, 10)

# Display top 10 PageRank results
alejandro_top_bigrams_df <- data.frame(word = names(alejandro_top_bigrams), rank = alejandro_top_bigrams)
print(alejandro_top_bigrams_df)

# Plotting Top 10 PageRank Bigrams for Alejandro
ggplot(alejandro_top_bigrams_df, aes(x = reorder(word, rank), y = rank, fill = word)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 PageRank Bigrams in Alejandro Comments",
       x = "Bigram",
       y = "PageRank Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
```

```r
reddit_comments_clean <- na.omit(ifelse(reddit_comments == "", NA, reddit_comments))

# Convert cleaned comments into a data frame
reddit_comments_df <- data.frame(text = reddit_comments_clean)

# Generate bigrams and filter out stopwords
reddit_bigrams <- reddit_comments_df |>
  unnest_tokens(output = bigram, input = text, token = "ngrams", n = 2) |>
  separate(bigram, into = c("word1", "word2"), sep = " ", extra = "drop", fill = "right") |>
  filter(!word1 %in% stopwords("en") & !word2 %in% stopwords("en") & !is.na(word1) & !is.na(word2))

# Count the frequency of each bigram
reddit_bigram_counts <- reddit_bigrams |>
  count(word1, word2, sort = TRUE) |>
  filter(n > 1)  # Optional: filter to focus on more frequent bigrams

# Create a graph from the bigram data
reddit_bigram_graph <- graph_from_data_frame(reddit_bigram_counts)

# Simplify the graph to remove loops and multiple edges
reddit_bigram_graph <- simplify(reddit_bigram_graph)

# Plot the bigram network using igraph
plot(reddit_bigram_graph, vertex.size = 4, edge.arrow.size = 0.5,
     main = "Bigram Network in Reddit Comments")

# Calculate PageRank
rank_reddit_bigrams <- sort(page_rank(reddit_bigram_graph)$vector, decreasing = TRUE)

# Get the top 10 PageRank scores
reddit_top_bigrams <- head(rank_reddit_bigrams, 10)

# Display top 10 PageRank results
reddit_top_bigrams_df <- data.frame(word = names(reddit_top_bigrams), rank = reddit_top_bigrams)
print(reddit_top_bigrams_df)

# Plotting Top 10 PageRank Bigrams for Reddit
ggplot(reddit_top_bigrams_df, aes(x = reorder(word, rank), y = rank, fill = word)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  labs(title = "Top 10 PageRank Bigrams in Reddit Comments",
       x = "Bigram",
       y = "PageRank Score") +
  theme_minimal() +
  scale_fill_brewer(palette = "Set3")
```

Figure 35 R code for "Alejandro" and "Reddit thread" Semantic Network

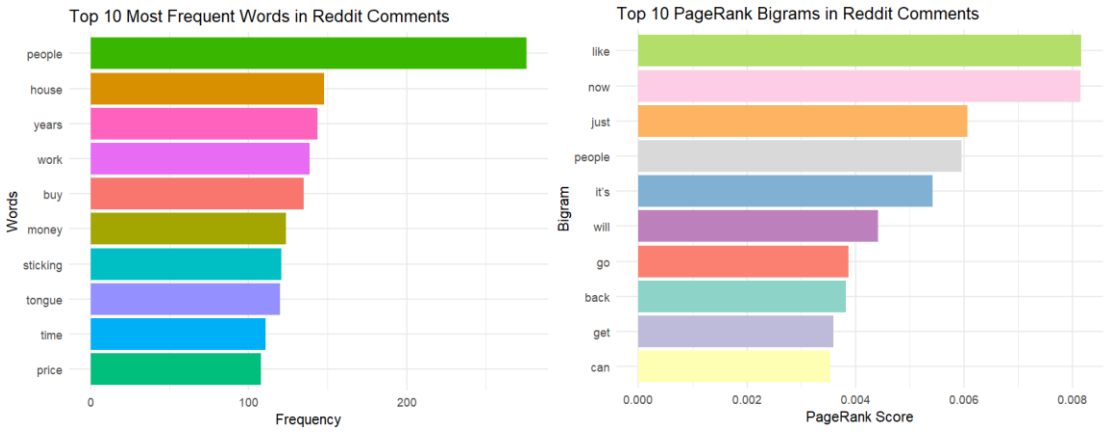


Figure 36 Comparison of Term Frequencies and PageRank Bigrams in Reddit Comments for "Meat Dress"

The comparison shows that while frequent words provide a snapshot of key topics, PageRank bigrams highlight the linguistic connections that give depth to the discussions.

Frequent Words: The most frequent words highlight the primary topics and themes discussed by commenters. Words like "people," "house," "years," and "work" indicate broader societal discussions, while terms like "buy," "money," and "price" reflect economic concerns. These words show the general content and focus of the comments.

PageRank Bigrams: The top PageRank bigrams reveal the structural importance of terms within the network of words used in the comments. Words like "like," "now," "just," and "people" being highly ranked show their connectivity and relevance in forming meaningful discussions. The presence of terms like "it's," "will," "go," "back," "get," and "can" suggests common linguistic structures used to frame thoughts, opinions, and actions.

The reason the rankings change so much is because the 2 data are processed by different mechanisms. ‘Frequency word’ just count how often each word appears in the comments. Frequently used words are ranked higher even if they are in different contexts. For example, ‘people’ and ‘work’ are frequently occurring common words. And PageRank takes into account the connections between words in the comment network. It identifies the core words in a discussion, not just the frequently occurring ones. Words like ‘like,’ ‘now,’ and ‘just’ may not be the most frequent, but they are highly correlated in the context of the comments, indicating their importance in the flow of dialogue.

More specifically, a high-frequency word like ‘people’, which is ranked first, is only

ranked fourth in PageRank, mainly because it is not as contextually relevant as ‘like’, ‘now’ and ‘just’. ‘and “just”. Overall, with this comparison chart, we can see that the topic of ‘Meat Dress’ is not only very controversial, but also introduces topics including but not limited to society, money, ideas, etc., and the commenters' opinions are proactive and subjective.

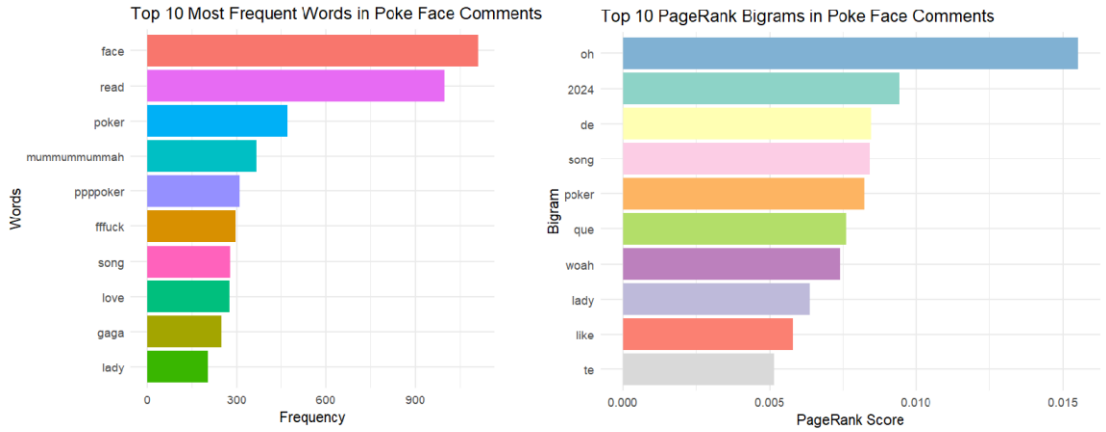


Figure 37 Comparison of Term Frequencies and PageRank in Comments on "Poker Face" Videos

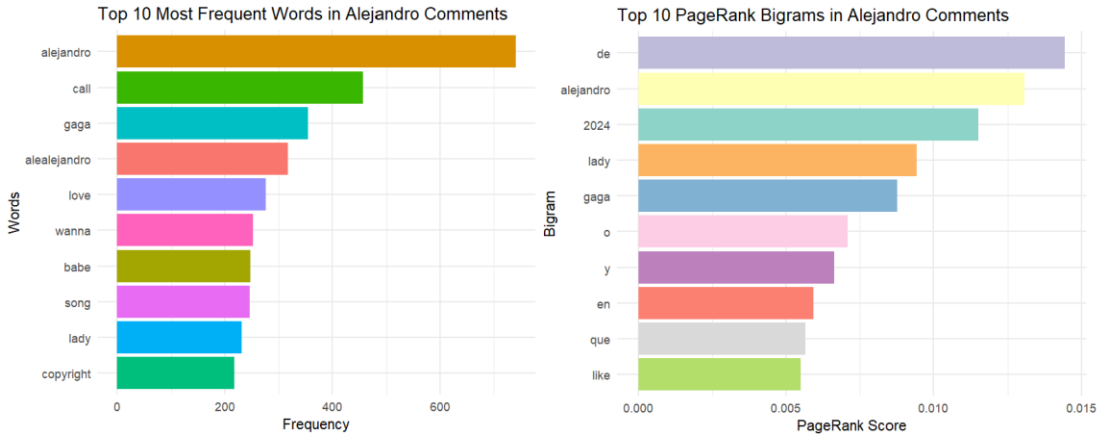


Figure 38 Comparison of Term Frequencies and PageRank in Comments on "Alejandro" Videos

In ‘Poker Face’, the bigrams "oh," "2024," and "woah" might not be the most frequent but are pivotal in the comment threads. This indicates that these words are part of important discussions or phrases within the network of comments. For example, "oh" might be part of expressions or quotes from the song, And in both of the videos, while the fact that ‘2024’ has high page rank means that both videos are very topical and are still being discussed even today.

It is worth noting that words like ‘de’ and ‘te’ as well as single letter words like ‘o’ and ‘y’ appear and have a high PageRank, may be due to problems in data preprocessing

or cleaning. It is possible that it is a leftover from not removing emoji or HTML code correctly, but by itself it is not detected as a deactivated word to be removed and the bit is more towards the centre of the sentence, so it has a high score in the calculation of page rank.

4.Social Network Analysis

4.1.1 Centrality Analysis

Degree centrality analysis

```
yt_actor_graph <- readRDS("PokeFaceActor.rds")
#reddit_actor_graph <- readRDS("RedditActor.rds")

# Find all maximum components that are weakly connected
yt_comps <- components(yt_actor_graph, mode = c("weak"))
yt_largest_comp <- which.max(yt_comps$csize)
yt_comp_subgraph <- yt_actor_graph |> induced_subgraph(vids = which(yt_comps$membership == yt_largest_comp))

# Degree Centrality
yt_degree_in <- sort(degree(yt_comp_subgraph, mode = "in"), decreasing = TRUE)[1:20]
yt_degree_out <- sort(degree(yt_comp_subgraph, mode = "out"), decreasing = TRUE)[1:20]
yt_degree_total <- sort(degree(yt_comp_subgraph, mode = "total"), decreasing = TRUE)[1:20]
```

Figure 39 R code for degree analysis for Poke Face

```
# Degree Centrality
alejandro_degree_in <- sort(degree(alejandro_comp_subgraph, mode = "in"), decreasing = TRUE)[1:20]
alejandro_degree_out <- sort(degree(alejandro_comp_subgraph, mode = "out"), decreasing = TRUE)[1:20]
alejandro_degree_total <- sort(degree(alejandro_comp_subgraph, mode = "total"), decreasing = TRUE)[1:20]
```

Figure 40 R code for degree analysis for Alejandro

Here we read the R data file and store the graph object in yt_actor_graph, and then use different modes of the degree function to calculate "degree in", "degree out" and “total degree” centrality respectively, and take out the first 20 of them.

Username	Score
VIDEOID:bESGLojNYSo	3001
@mahdihaidari2446	26
@Amanteigualitario	22
@GloeJassie	20
@tonys.5114	20
@ahyuyuyu6949	20
@streetsmart.69	20
@NicoDR_BS	20
@user-kl7vd6jy8c	20
@jhaber.2024..	20
@_____NoComment.....	20
@chicken	20
@louiseking5655	19
@Nellia.20x	11
@Triple87	11
@Samariapain	10
@MeR-md1jq	10
@sanar983	9
@wsopzork	8
@pedroruiz3943	8

Username	Score
VIDEOID:niqrrmev4mA	3001
@LexiMarilyn	20
@latinoburger123	20
@NicoDR_BS	20
@mahdihaidari2446	20
@beatrzclemente8341	20
@dalsian1548	20
@phillipsosa3353	19
@Xiao._522	14
@Minho_969	13
@lostvibes5043	13
@Politischinkorekt	10
@AntonioOrtiz1	10
@sumeetkumarbordoloi4512	9
@MissMegiable	9
@erikmayeku7003	8
@lmPeely	8
@raffausfaus	7
@YazminFarraj-ij9wt	7
@user-mj5md4me3g	6

Figure 41 In-Degree Centrality on Poker Face (left) and Alejandro (right)

In-degree centrality measures the number of incoming connections to a node. The highest-ranked nodes on both sides are the videos themselves, which is expected for YouTube, as all comments are directed at and made under the video. Therefore, this doesn't represent too much meaning. However, aside from that, we can see that the level of in-degree centrality for the two videos is relatively close, but "Poker Face" is slightly higher. Users like @mahdihaidari2446 and @Amanteigualitario have scores of 26 and 22, respectively. These higher scores indicate the popularity or importance of these users in the network, or the importance of the topic. Users with slightly lower scores, like @user-mj5md4me3g, indicate that they are less central to the discussion.

Username	Score
@mariaguadalupesanchezgonza8023	37
@wsopzork	33
@DCh-qy5vo	25
@msmalik529	23
@joaquinvaleri7022	19
@Samariapain	17
@Amanteigualitario	15
@Mitchell710sdg	14
@frankieL1882	12
@vivewrgaingpower7747	12
@user-qp1ft6fk6q	9
@jmc5368	8
@zeeshanzohaib905	7
@SarahMargaretMeseberg	7
@rishabhgautam2723	6
@bennyBremixdj	6
@mary-janerock4332	6
@gotstrange1	6
@larrychuchua6287	6
@MassimilianoMarotta-sc3ci	5

Username	Score
@user-vm7kq7po8j	12
@voyagerone7487	10
@MissMegiable	10
@Beatriz-tb5gk	9
@dumitrumorosanu9543	9
@mikoaipan7351	8
@luissarinana7816	8
@user-vo6dq4br7i	8
@LPRpgler25	7
@camarada6456	7
@AntonioAngelini-ks4om	7
@Moonlight-uv4mu	6
@darkjudazer	6
@MarioMendoza-ic9my	6
@arissoooc	6
@BlisaBLisa	6
@Politischinkorekt	6
@sylwiahusar1101	6
@stephanielee3491	6
@Deedee0088	6

Figure 42 Out-Degree Centrality on Poker Face (left) and Alejandro (right)

Out-degree centrality measures the number of outgoing connections from a node. It indicates how many comments a user has made. We can see that Poker Face's overall score is higher than Alejandro's overall score, which means that Poker Face's users' desire to output is higher than Alejandro's users', the more frequently the same user inputs, the more it shows that he is more interested in this video. This suggests that Poker Face users are more active, possibly because they have strong opinions or want to participate in many discussions. On the other hand, Alejandro's users have a low score, which may indicate that Alejandro's users have a low desire to participate in the discussion, most likely because the video does not generate a lot of discussion.

An interesting point is that in 'Alejandro', the @NicoDR_BS user has an In-degree of 20, but is not on the Out-degree list, which suggests that they may not have a lot of comments, but are more likely to be discussed or popular. Users like @Amanteigualitario from 'Poke Face' have In-degree 22 and Out-degree 15, which suggests that they are not only active in the comments, but are often mentioned and discussed.

Username	Score
VIDEOID:bESGLojNYSo	3002
@wsopzork	41
@mariaguadalupesanchezgonza8023	39
@Amanteigualitario	37
@DCh-qy5vo	29
@mahdihaidari2446	28
@Samariapain	27
@msmalik529	23
@louiseking5655	22
@jhaber.2024..	22
@GloeJassie	21
@tonys.5114	21
@ahyuyuyu6949	21
@streetsmart.69	21
@NicoDR_BS	21
@user-kl7vd6jy8c	21
@_____NoComment.....	21
@chicken	21
@joaquinvaleri7022	19
@Mitchell710sdg	14

Username	Score
VIDEOID:niqrrmev4mA	3002
@LexiMarilyn	21
@latinoburger123	21
@NicoDR_BS	21
@mahdihaidari2446	21
@beatrzclemente8341	21
@phillipsosa3353	21
@dalsian1548	21
@MissMegiable	19
@Xiao._522	18
@Politischinkorekt	16
@Minho_969	14
@lostvibes5043	14
@user-vm7kq7po8j	12
@camarada6456	12
@sumeetkumarbordoloi4512	12
@AntonioOrtiz1	11
@erikmayeku7003	10
@Anonymous-zw9ud	10
@voyagerone7487	10

Figure 43 Degree Total Centrality on Poker Face (left) and Alejandro (right)

Degree total centrality measures the overall importance of a node by combining its in-degree and out-degree centralities, which indicate how many connections it has incoming and outgoing. In the "Poker Face" dataset, the top users have significantly higher total degree centrality scores, with the highest being 41, compared to 21 in the "Alejandro" dataset. The "Poker Face" dataset shows a wider range of total degree centrality scores, with several users having scores above 20. In contrast, the "Alejandro" dataset has a more compressed range, with the highest user scores also around 21. This indicates that users in the "Poker Face" dataset are generally more active in terms of both commenting and being mentioned, suggesting higher overall engagement in this network.

It is noteworthy that @mahdihaidari2446 and @Amanteigualitario have high Degree Total Centrality in both videos. This suggests that these users are likely loyal fans of Lady Gaga. Their high engagement across multiple videos implies a consistent and significant level of interaction with her content. Loyal fans like @mahdihaidari2446 and @Amanteigualitario can influence other viewers' perceptions and interactions. Their frequent mentions and active participation can help sustain discussions and keep the community engaged.

Degree centrality analysis

```
# Closeness Centrality
yt_closeness_in <- sort(closeness(yt_comp_subgraph, mode = "in"), decreasing = TRUE)[1:20]
yt_closeness_out <- sort(closeness(yt_comp_subgraph, mode = "out"), decreasing = TRUE)[1:20]
yt_closeness_total <- sort(closeness(yt_comp_subgraph, mode = "total"), decreasing = TRUE)[1:20]
```

Figure 44 R code for closeness centrality analysis for Poker Face

```
# Closeness Centrality
alejandro_closeness_in <- sort(closeness(alejandro_comp_subgraph, mode = "in"), decreasing = TRUE)[1:20]
alejandro_closeness_out <- sort(closeness(alejandro_comp_subgraph, mode = "out"), decreasing = TRUE)[1:20]
alejandro_closeness_total <- sort(closeness(alejandro_comp_subgraph, mode = "total"), decreasing = TRUE)[1:20]
```

Figure 45 R code for closeness centrality analysis for Alejandro

Refer to degree centrality code, here we use “closeness” function instead of “degree” to find the closeness centrality for both videos.

Username	Score
@DCh-qy5vo	1
@gulden86209	1
@nassimnaitkhouya9893	1
@schelleyl.7263	1
@wsopzork	1
@yugibros1	1
@billz9303	1
@CamiloAguilarRios	1
@YungJora	1
@brittanymullins7482	1
@solangerosendo	1
@JavierHernandez-mk2in	1
@BrendaRamos-if1dh	1
@vro5225	1
@ngabershayyuk	1
@e-10100	1
@HungPham-wx2qr	1
@iDeathYT	1
@MejganZia	1
@arunmohanan7786	1

Poker Face, In

Username	Score
@heatherkeith6934	1
@Love-LGTSDLAGTSCP	1
@abdullahalmamun4499	1
@MrKeybladem aster1992	1
@Mitchell710sdg	1
@Mr.matos1999	1
@lilastar2566	1
@bashkurd2769	1
@rufuspdoggie3262	1
@UlulAlbab-sn8zn	1
@marioalberto4972	1
@claydillard3563	1
@SarahKoo_Piano	1
@user-tp9wk5rz3d	1
@rafaelgonzales130	1
@blastisiafondaria9906	1
@MiliE_FerOFFICIAL	1
@ManggouManggouthangew	1
@awakwd1987	1
@LordFufuu	1

Poker Face, Out

Username	Score
@naythithtinaung5379	1
@blessingsmwale4546	1
@MadsonJoao	1
@omarperez8195	1
@Gasimova155	1
@user-fy5rf4gw7o	1
@portela-gb1mt	1
@VitorGabriel-sb3pd	1
@PINOYVIRALS-PH	1
@luisaluveniyali8962	1
@anjelodoleo7643	1
@LudmilaP-wi4sk	1
@xXEltumbaburros125xX	1
@alejandroruiz8495	1
@user-ww6lw8og3f	1
@tiberiunisipasu8657	1
@esdras795	1
@fatimaresendis3981	1
@shawngall7846	1
@rozehad	1

Alejandro, In

Username	Score
@guydavid2580	1
@israeluwu6942	1
@AH_GAMING-ps6nx	1
@naviiClips	1
@johangielen128	1
@antoniomarruecos9429	1
@SaraiDiaz-qi7gi	1
@eliseoestrada5371	1
@MarisolPerezMota	1
@RadomirRomanov	1
@user-iq8xj3jn1q	1
@user-pg1li9xw2u	1
@AlbertAminov	1
@agustdlvu	1
@prettygae29	1
@eliaslahab1994	1
@umshitposterqualquer1637	1
@paulojose6791	1
@lilliannaly118	1
@IA-uk4hk	1

Alejandro, Out

Figure 46 In and Out closeness centrality for “Poker Face” and “Alejandro”

The closeness centrality values for both "Closeness In Centrality" and "Closeness Out Centrality" are 1 for all users in the YouTube videos. This uniformity can be attributed to specific characteristics of the network structure and how closeness centrality is computed in this context. In the context of YouTube video comments, the network might be considered as a star topology, where the video itself is the central node, and all comments are directly connected to this node. In such a topology, the distance from any comment node to any other comment node is the same, making the network effectively equidistant for all nodes.

Username	Score
VIDEOID:bESGLojNYSo	0.0002912904
@mahdihaidari2446	0.0001566416
@jhaber.2024..	0.0001565435
@streetsmart.69	0.0001564456
@_____NoComment.....	0.0001564456
@NicoDR_BS	0.0001563966
@user-kl7vd6jy8c	0.0001563966
@chicken	0.0001563722
@ahyuyuyu6949	0.0001563477
@Amanteigualitario	0.0001562500
@GloeJassie	0.0001562012
@louiseking5655	0.0001561768
@rishabhgautam2723	0.0001561037
@tonys.5114	0.0001561037
@longyone1980	0.0001560549
@bruno8286	0.0001560549
@szentesidavid7447	0.0001560306
@subushiv	0.0001559333
@AoibhinnsBraceletsLove	0.0001559333
@MeR-md1jq	0.0001559333

Username	Score
VIDEOID:niqrrmev4mA	0.0002955083
@Xiao._522	0.0001567890
@beatrzclemente8341	0.0001567152
@mahdihaidari2446	0.0001565190
@dalsian1548	0.0001565190
@LexiMarilyn	0.0001564945
@latinoburger123	0.0001564945
@phillipsosa3353	0.0001564945
@NicoDR_BS	0.0001562988
@Minho_969	0.0001561524
@lostvibes5043	0.0001561524
@Lux-ue1md	0.0001560549
@TomsTrailerReactions	0.0001560549
@arissoooc	0.0001560306
@AntonioOrtiz1	0.0001560306
@Nawafz13	0.0001560062
@yuttsuun	0.0001559819
@OritAnastasiaChannel	0.0001559576
@lmPeely	0.0001559576
@camarada6456	0.0001559333

Figure 47 Closeness Total Centrality on Poker Face (left) and Alejandro (right)

High closeness centrality indicates that a user is close to all other users in the network, suggesting they can quickly interact with or influence many others. Refer to degree centrality of two videos, @mahdihaidari2446 and @NicoDR_BS both appear with high scores. And now,@mahdihaidari2446 and @NicoDR_BS still appear with high closeness total centrality in both "Poker Face" and "Alejandro" datasets. This suggests that these users are highly influential and well-connected in the network for both videos. Their ability to quickly interact with many other users makes them key figures in maintaining active and lively discussions.They can quickly spread information and engage with a wide audience, indicating they are likely dedicated fans of Lady Gaga.

Betweenness centrality analysis

```
# Betweenness Centrality poker face
yt_betweenness <- sort(betweenness(yt_comp_subgraph, directed = FALSE), decreasing = TRUE)[1:20]
# Betweenness Centrality alejandro
alejandro_betweenness <- sort(betweenness(alejandro_comp_subgraph, directed = FALSE), decreasing = TRUE)[1:20]
```

Figure 48 R code for betweenness centrality analysis for both videos

Refer to degree centrality code, here we use “betweenness” function instead of “degree” to find the betweenness centrality for both videos.

Username	Score
VIDEOID:bESGLojNYSo	4497845.76
@mahdihaidari2446	71724.00
@NicoDR_BS	56829.00
@user-kl7vd6jy8c	54094.92
@_____NoComment.....	53849.00
@chicken	53847.50
@ahyuyuyu6949	53847.00
@Amanteigualitario	44629.25
@jhaber.2024..	41929.50
@GloeJassie	41014.11
@louiseking5655	37989.97
@tonys.5114	37582.87
@streetsmart.69	32952.33
@Nellia.20x	26964.00
@sanar983	26964.00
@MeR-md1jq	25465.50
@pedroruiz3943	22475.00
@Triple87	18584.69
@marilyndeservedbetter	16482.50
@el_tunometecabra7171	14990.00

Username	Score
VIDEOID:niqrrmev4mA	4631542.096
@LexiMarilyn	57665.000
@mahdihaidari2446	57665.000
@dalsian1548	57665.000
@latinoburger123	53623.595
@NicoDR_BS	45555.000
@beatrzclemente8341	44073.226
@Minho_969	36462.000
@lostvibes5043	36462.000
@Xiao._522	30511.048
@phillipsosa3353	29265.225
@lmPeely	22795.295
@AntonioOrtiz1	22481.065
@sumeetkumarbordoloi4512	12880.624
@raffausfaus	12602.333
@erikmayeku7003	12170.500
@mayconavarrete6857	12170.000
@josephosullivan9891	12170.000
@endofaneraoutnow3473	9558.833
@AuthorRoseKnight	9129.000

Figure 49Closeness Centrality on Poker Face (left) and Alejandro (right)

Video Nodes: Both video nodes have extremely high betweenness centrality scores, which is expected since all user comments are linked to these videos. Users like @mahdihaidari2446, @NicoDR_BS, @Amanteigualitario, and @LexiMarilyn have high betweenness centrality scores in both videos, indicating their significant role in connecting other users and facilitating interactions within the network. These users can significantly impact the dynamics of the community. Their central position allows them to influence conversations, mediate conflicts, and introduce new topics or trends.

The distribution of betweenness centrality scores shows that a few key users hold significant influence in the network, while the majority of users have lower scores. Users with high betweenness centrality in both videos, such as @mahdihaidari2446 and @NicoDR_BS, likely represent loyal fans of Lady Gaga. Their consistent engagement and influence across multiple videos highlight their dedication and importance in the fan community.

4.2 Community Analysis

Louvain method

```r
#load 2 videos and 1 reddit as network graphs
yt1_actor_graph <- readRDS("PokeFaceActor.rds")
yt2_actor_graph <- readRDS("AlejandroActor.rds")
reddit_actor_graph <- readRDS("RedditActor.rds")

#Transform to Undirected Graphs
yt1_undir_graph <- as.undirected(yt1_actor_graph, mode = "collapse")
yt2_undir_graph <- as.undirected(yt2_actor_graph, mode = "collapse")
reddit_undir_network_graph <- as.undirected(reddit_actor_graph, mode = "collapse")


# Louvain Method--------------------------------------------------------------

#YouTube Dataset 1: Poker Face:
# Apply Louvain algorithm
yt1_louvain_comm <- cluster_louvain(yt1_undir_graph)
# View the sizes of the communities
sizes(yt1_louvain_comm)
# Visualize the Louvain communities
plot(yt1_louvain_comm,
     yt1_undir_graph,
     vertex.label = V(yt1_undir_graph)$name,
     vertex.size = 4,
     vertex.label.cex = 0.7)


#YouTube Dataset 2: Alejandro:
# Apply Louvain algorithm
yt2_louvain_comm <- cluster_louvain(yt2_undir_graph)
# View the sizes of the communities
sizes(yt2_louvain_comm)
# Visualize the Louvain communities
plot(yt2_louvain_comm,
     yt2_undir_graph,
     vertex.label = V(yt2_undir_graph)$name,
     vertex.size = 4,
     vertex.label.cex = 0.7)


#Reddit Dataset:
reddit_louvain_comm <- cluster_louvain(reddit_undir_network_graph)
sizes(reddit_louvain_comm)
plot(reddit_louvain_comm,
     reddit_undir_network_graph,
     vertex.label = V(reddit_undir_network_graph)$name,
     vertex.size = 4,
     vertex.label.cex = 0.7)
###NOTE: the plot graph very hard to see, i will show it in graphic######
```

Figure 50 R code for Louvain method

```
Community sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   25   26   27   28
2278    7   12    2    2   44    1    2   38    2    2    2   46    2    2    6    2    2    3    1    2    2    1    3    2    2    2    2
  29   30   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54   55   56
   2    2    2    2    1    2    3    3    1    2    3    2    3    2    5    2    5    2    2    2    4   20    2    6   39    1    2    2
  57   58   59   60   61   62   63   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80   81   82   83   84
   3    2    3   10    2    2    2    1    2    2    3    2    4    2    2    2   10    4    2    2    2    5    2    2    1    1    2    2
  85   86   87   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102  103  104  105  106  107  108  109  110  111  112
   2    2    3    2    2    2    2    2    1    1    6    4    2    2    2   21    2    1    3    2   19    2    4    2    2    3    2    2
 113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140
   3    2    2    2    3    4    2    1    2    2    2    4    2    2   26    4    2    2    2    2    1    2    2    2    2    2    2    2
 141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168
   1    2    2    5    3    2    2    1    2   24    2    1    2    2    4    5    2    2    1    4    2    2    2    2    3    2    2    2
 169  170  171  172  173  174  175  176  177  178  179  180  181  182  183
   2    7    3    3    2    5    2    2    2    2    1    2    2    1    2
> #YouTube Dataset 2: Alejandro:
> # Apply Louvain algorithm
> yt2_louvain_comm <- cluster_louvain(yt2_undir_graph)
> # View the sizes of the Alejandro communities
> sizes(yt2_louvain_comm)
Community sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   25   26   27   28
2442   60   59    3    2    2    2    2    2    1    2    6    2    2   20    3    3    1   32    2    2    2    2   13    1    2    4    2
  29   30   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54   55   56
   2    3    2    2    4   13    2    2    2    2    1    2    2    8    2    2    2    2    3    1   21   16    2    2    2    2    1    2
  57   58   59   60   61   62   63   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80   81   82   83   84
   1    6    2    2    3    2    1    2    2    2   32    3    2    2    3    2    2    1    2    1    2    3    2    3    2    2    2    7
  85   86   87   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102  103  104  105  106  107  108  109  110  111  112
   2    1    2    2    2    2    2    3    2    2    4    2    3    2    2    2    2    2    2    2    2   21    2    2    3    3    2    2
 113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140
   2    2    1    2    2    2    2    2    2    2    3    2    2    1    2    2    2    2    2    2    2    4    2    2    2    3    2    3
 141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156
   2    2    2    2    2    2    2    2    2    2    1    1    3    1    2
> #Reddit Dataset:
> reddit_louvain_comm <- cluster_louvain(reddit_undir_network_graph)
> sizes(reddit_louvain_comm)
Community sizes
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
16 23 64 17 38 37 28 19 23 18 13 24  9 16  8 19 47 10  6
```

Figure 51 result of Louvain method on Poker Face (top), Alejandro (mid) and Reddit (bottom)

The Louvain method, a community detection algorithm, optimizes modularity to identify communities within a graph. This approach is efficient and widely used for large networks. Here, we analyze the community sizes and visualize the results for three datasets: "Poker Face", "Alejandro" and a Reddit dataset.

For the "Poker Face" dataset, the Louvain method reveals a dominant community with 2278 members. Other communities are significantly smaller, with sizes ranging from 2 to 60 members. This indicates a central hub of interactions for the "Poker Face" video. In the other hand, the analysis of the "Alejandro" dataset shows a similar pattern, with the largest community comprising 2442 members. Other communities range from 2 to 59 members. This confirms the presence of a central community with many users, surrounded by smaller isolated groups. The Reddit dataset presents a more varied distribution of community sizes. The largest community has 64 members, and other communities have sizes ranging from 2 to 23 members. This pattern is more balanced compared to the YouTube datasets, suggesting diverse interaction dynamics within the Reddit thread.

However, in the Louvain method community detection analysis for the "Poker Face" and "Alejandro" datasets, it is noteworthy that in the "Alejandro" dataset, communities 2 and 3 have relatively high member counts, with 60 and 59 members respectively. This contrasts with the "Poker Face" dataset, which does not exhibit a similar pattern. Several explanations for this phenomenon are as follows:

1. **Controversial Content:**
   The theme, style, or content of "Alejandro" might have generated higher interaction interest among specific audience groups. This interest could stem from the audience's positive reception of the video or from controversial topics arising from the video. These topics may have engaged related audiences, forming closely-knit interactive groups in the comments section. For instance, the distinct visual and musical elements of "Alejandro" could spark extensive discussions, leading to the formation of larger communities. The unique artistic direction, cultural references, or even the narrative of the video could be points of contention or appreciation, driving more comments and replies among viewers.

2. **Comment Timing and Hotspot Events:**
   There could have been particular time periods during which the "Alejandro" video experienced increased viewer attention due to related events. These could include music festivals, media reports, or trending discussions on social media platforms. Such events might trigger significant discussions in the comments section, resulting in the formation of larger interactive communities. If, for example, Lady Gaga performed "Alejandro" at a major event or if the song was featured in popular

media, it could lead to a surge in comments, thereby creating substantial communities of interaction.

3. **Cross-Interaction Among User Groups:**
   There might be highly active user groups participating in multiple communities within the "Alejandro" dataset. These users might not only be engaged in community 1 but also actively participate in discussions within communities 2 and 3. The cross-interaction of these user groups could lead to multiple active communities, resulting in the formation of several larger communities. This indicates that certain users have a broad influence and interact with various segments of the audience, enhancing the overall engagement within the video's comment section.

4. **Algorithm Detection Sensitivity:**
   The Louvain method clusters users based on the frequency and patterns of their interactions. In the "Alejandro" dataset, it is possible that there are multiple groups of users who frequently interact with each other, leading to the formation of larger communities. The detection algorithm identifies these frequent interactions and groups these users together, resulting in multiple large communities.

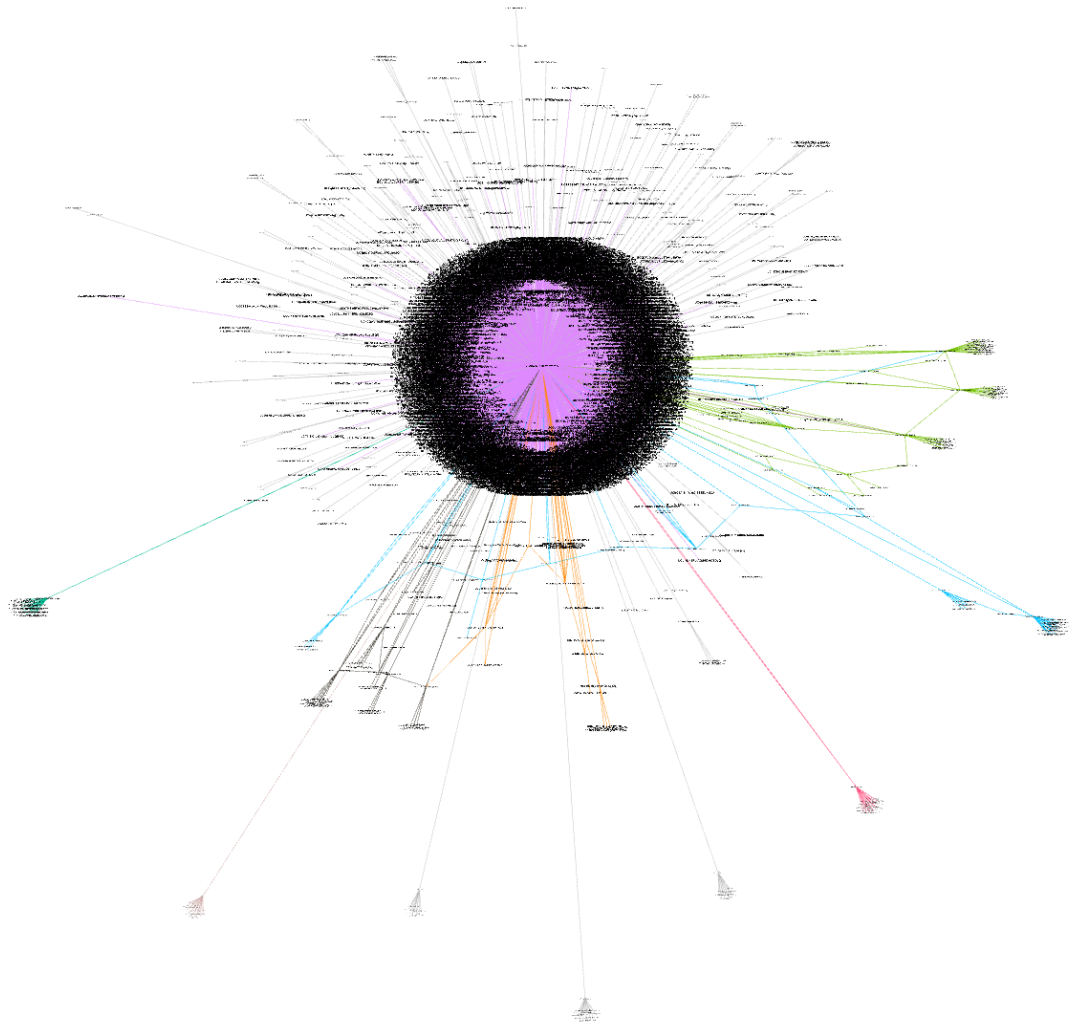


Figure 52 Gephi, network of actors on the "Alejandro"

The above graph represents the network of comments on the "Alejandro" video from Gephi network analysis tool. In this visualization, nodes represent individual users, and edges represent interactions between them, such as replies to comments. The purple color indicates a dense cluster of interactions at the core, suggesting a highly interconnected community. The various colors of the outer nodes suggest smaller sub-communities or clusters of interactions. Despite the application of the modularity algorithm, the high level of interconnectivity makes it challenging to discern distinct communities clearly.

Refer back to Louvain algorithm, which identified communities of varying sizes, from a large community with 2,442 members to several smaller ones with only one or two members. The network graph applied to the "Alejandro" dataset reveals a diverse and uneven distribution of community sizes, which aligns with the results obtained from the Louvain algorithm analysis.

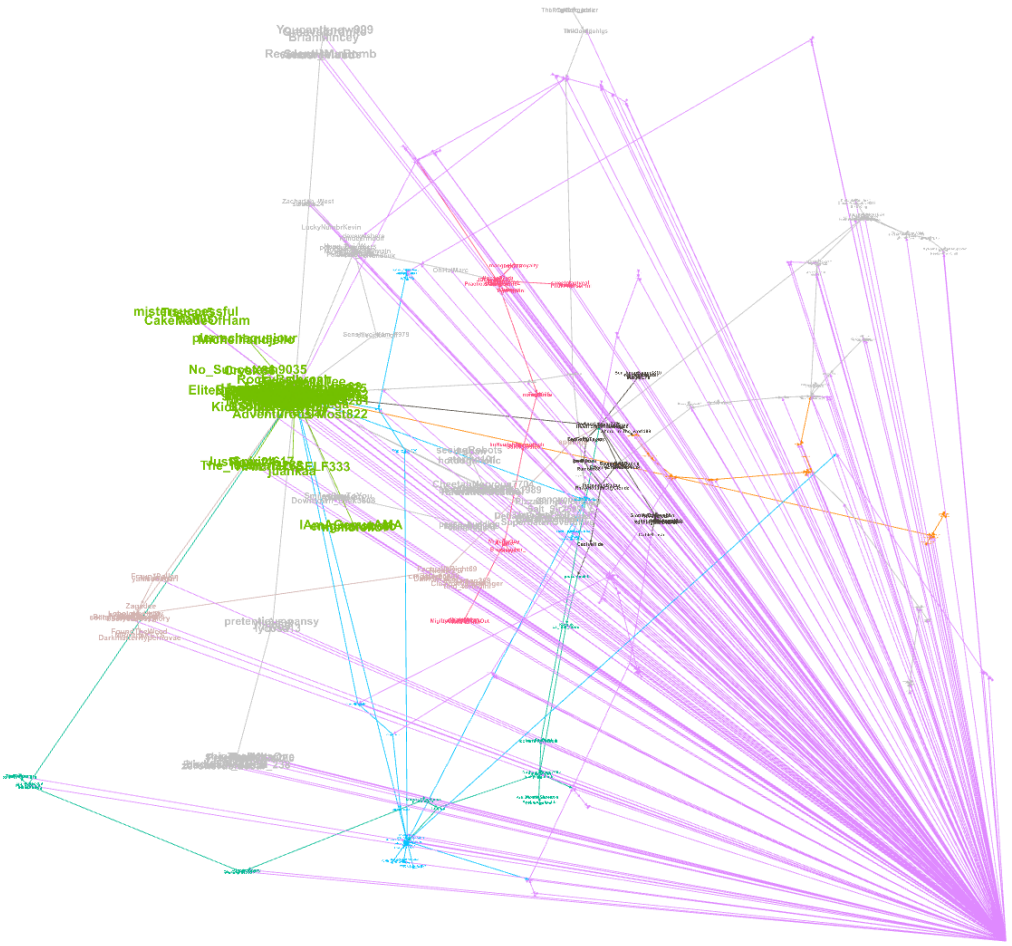


Figure 53 Gephi, network of actors on Reddit

The second graph represents the Reddit discussion about Lady Gaga's meat dress. This network also shows nodes as users and edges as interactions. Here, the modularity algorithm more clearly identifies distinct clusters, indicated by different colors. Each color represents a community of users who interact more frequently with each other than with users in other clusters. This suggests that the Reddit discussion is more segmented, with users forming tighter-knit groups around specific subtopics or viewpoints related to the meat dress discussion.

Refer back to Louvain algorithm, the network graph also reveals a distinct and segmented community structure, consistent with the Louvain algorithm results. The graph shows multiple small to medium-sized communities, indicating a more decentralized discussion pattern. The largest community contains 64 members, while many other communities have fewer members, ranging from a few to several dozen. This segmentation reflects the modularity visualization, where distinct clusters or communities were identifiable, each possibly focusing on different aspects or viewpoints of the topic.

Girvan-Newman method

```r
#YouTube Dataset 1: Poker Face:
# Apply Girvan-Newman algorithm
yt1_gn_comm <- cluster_edge_betweenness(yt1_undir_graph)
# View the sizes of the communities
sizes(yt1_gn_comm)
# Visualize the Girvan-Newman communities
plot(yt1_gn_comm,
     yt1_undir_graph,
     vertex.label = V(yt1_undir_graph)$name,
     vertex.size = 4,
     vertex.label.cex = 0.7)


#YouTube Dataset 2: Alejandro:
# Apply Girvan-Newman algorithm
yt2_gn_comm <- cluster_edge_betweenness(yt2_undir_graph)
# View the sizes of the communities
sizes(yt2_gn_comm)
# Visualize the Girvan-Newman communities
plot(yt2_gn_comm,
     yt2_undir_graph,
     vertex.label = V(yt2_undir_graph)$name,
     vertex.size = 4,
     vertex.label.cex = 0.7)


#Reddit Dataset:
# Apply Girvan-Newman algorithm
reddit_gn_comm <- cluster_edge_betweenness(reddit_undir_graph)
# View the sizes of the communities
sizes(reddit_gn_comm)
# Visualize the Girvan-Newman communities
plot(reddit_gn_comm,
     reddit_undir_graph,
     vertex.label = V(reddit_undir_graph)$name,
     vertex.size = 4,
     vertex.label.cex = 0.7)

###NOTE: the plot graph very hard to see, i will show it in graphic######
```

Figure 54 R code for Girvan-Newman algorithm

```
> sizes(yt1_gn_comm)
Community sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   25   26   27
2342   17    2    2   15    2   25    6    2    2    4    2    3    2    2    2    9    2    5    2    2    2    2    3    2    3    2
  28   29   30   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54
   3    2    5    4    3    2    5    2   17   10    2    2    4   20    6   18    2    2    3    2    3   10    2    2    2    2    4
  55   56   57   58   59   60   61   62   63   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80   81
   2    2    2   10    4    2    2   33    5    2    2    2    2    2    3    2    2    2    2    2    6    4    2    2   21    2    3
  82   83   84   85   86   87   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102  103  104  105  106  107  108
   2   19    2    4    2    2    3    2   21    3    2    2    2    3    4    2    2    2    2   21    4    2   26    4    2    2    2
 109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135
   2    2    2    2    2    2    2    2    2    5    3    2    2   24    2    4    5    2    2    4    2    2    2    2    3    2    2
 136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152
   2    2    4    3    3    2    5    2    2    3    2    2    2    2    2    2    2
> # View the sizes of the communities
> sizes(yt2_gn_comm)
Community sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   25   26   27
2486    4   34    6    3    2    2    2    2    6    2    2   20    3    3    7    2    2    2   13    2    4    2    3    2    2    4
  28   29   30   31   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48   49   50   51   52   53   54
  13    2    2    4    2    2    2   27    6    2   13    2    3   21   16    2    2    2    2    2    6    2    2    3    2    2    2
  55   56   57   58   59   60   61   62   63   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80   81
  17    2    8    3    2   28    2    8    3    2    2    2    2    3    2    3    9    2    7    2    2    2    3    2    3    2    2
  82   83   84   85   86   87   88   89   90   91   92   93   94   95   96   97   98   99  100  101  102  103  104  105  106  107  108
   4    2    3    2    2    2    2    2    2    2    3    2   21    2    2    3    3    2    2    2    2    2    2    2    2    2    2
 109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135
   2    2    3    2    2    2    2    2    2    2    8    2    4    2    2    3    2    3    2    2    2    2    2    2    2    2    2
 136  137  138  139
   2    2    3    2
> # View the sizes of the communities
> sizes(reddit_gn_comm)
Community sizes
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
64 18  7  7  4 11  9 10 27  3  9 19  8 22 13  7  6  5 14 22  9 18 12  5  4  8 11 11 49 13  6  4
```

Figure 55 result of Girvan-Newman method on Poker Face (top), Alejandro (mid) and Reddit (bottom)

The Girvan-Newman method is a community detection algorithm, identifies communities within a graph based on the edge betweenness centrality. This approach iteratively removes edges with the highest betweenness centrality, resulting in the division of the graph into separate communities. Here, we analyze the community sizes and visualize the results for three datasets: "Poker Face", "Alejandro", and a Reddit dataset.

For the "Poker Face" dataset, the Girvan-Newman method reveals a dominant community with 2342 members. Other communities are significantly smaller, with most having only 2 to 4 members. This suggests that the "Poker Face" video has a central hub or central topic where a large number of users interact closely, while the remaining users form smaller, less connected groups.

The Alejandro dataset's largest community includes 2486 nodes, which is even larger than the Poker Face dataset, suggesting a higher level of centralization and engagement. Like the Poker Face dataset, there are also mid-sized communities ranging from 10 to 100 nodes and small communities of 2 to 5 nodes. The larger main community here indicates that the content of the Alejandro video might be generating more engagement, leading to a more centralized interaction pattern.

The Reddit dataset has a largest community with 64 nodes, which is significantly smaller than those found in the YouTube datasets. This smaller size suggests less centralization in user interactions, reflecting Reddit's nature of having more fragmented and topic-specific discussions. Mid-sized communities range from 7 to 22 nodes, and small communities consist of 4 to 11 nodes. This pattern highlights the diverse and less centralized nature of Reddit discussions compared to YouTube.

When comparing the Girvan-Newman results with those from the Louvain method, several similarities and differences emerge. Both methods identify a large main community and numerous smaller ones in the YouTube datasets. However, the Girvan-Newman algorithm tends to reveal more granular sub-communities, offering a finer breakdown of interactions.

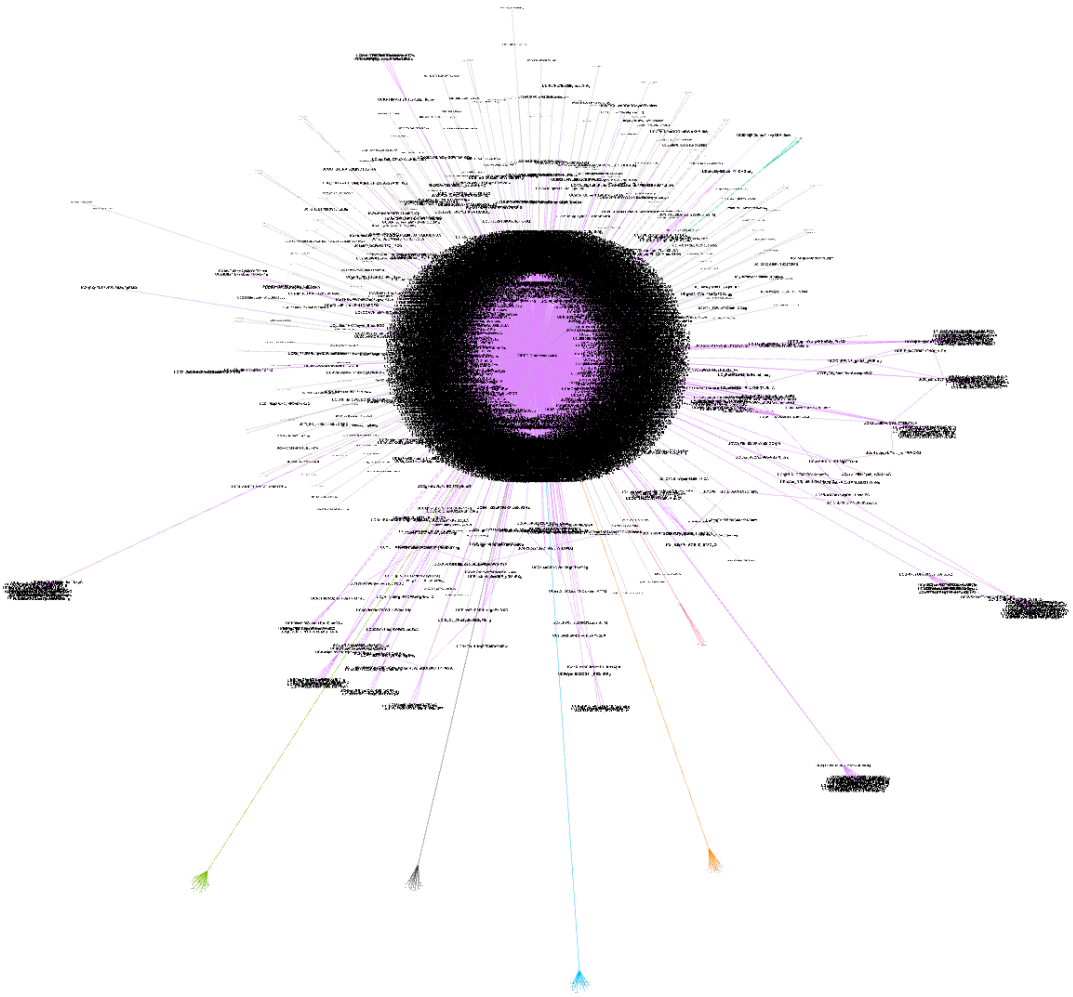


Figure 56 Gephi, network of comments with Girvan-Newman on the "Alejandro"

From the Girvan-Newman clustering results, we observe a similar trend with Louvain community. The community sizes for "Alejandro" range from 2 to 268 nodes, with the largest community having a considerable number of interactions compared to the smaller ones.

The structure of YouTube's comment system, where comments are directly linked to the video, inherently centralizes the interactions around the video itself. This is why, under both the Girvan-Newman and Louvain methods, the YouTube community detection results reveal one particularly large community with over 2000 nodes. This large community represents the bulk of interactions revolving directly around the video, which is reflected in the dense and centralized shape of the Gephi visualizations.

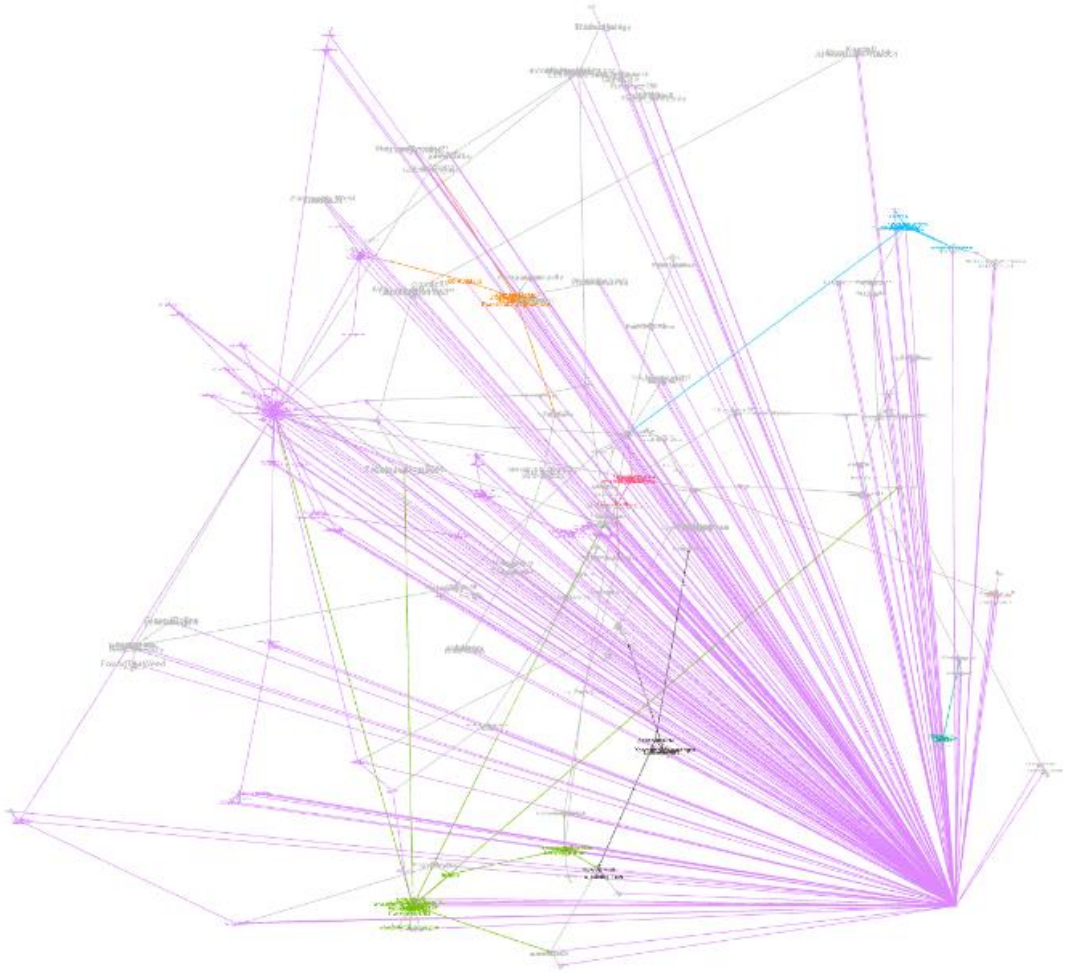


Figure 57 Gephi, network of comments with Girvan-Newman on Reddit

The Reddit network visualization shows a more dispersed structure compared to the "Alejandro" video. The nodes are spread out with distinct clusters, each representing a different community. The community sizes for the Reddit thread, as revealed by the Girvan-Newman method, range from very small clusters of 2 to 64 nodes. This spread indicates a wider variety of smaller, more focused discussions, as opposed to the centralized, large communities seen in the "Alejandro" video. This difference highlights the varied nature of Reddit, where multiple topics and sub-discussions can thrive independently within a single thread.

5. Machine Learning Models

5.1 Sentiment Analysis

```r
#Alejandro
clean_alejandro_text <- alejandro_yt_data$Comment |>
  replace_url() |>
  replace_html() |>
  replace_non_ascii() |>
  replace_word_elongation() |>
  replace_internet_slang() |>
  replace_contraction() |>
  removeNumbers() |>
  removePunctuation()

sentiment_alejandro_scores <- get_sentiment(clean_alejandro_text, method = "afinn") |> sign()
sentiment_alejandro_df <- data.frame(text = clean_alejandro_text, sentiment = sentiment_alejandro_scores)
sentiment_alejandro_df$sentiment <- factor(sentiment_alejandro_df$sentiment, levels = c(1, 0, -1),
                                           labels = c("Positive", "Neutral", "Negative"))

# Plot sentiment classification
ggplot(sentiment_alejandro_df, aes(x = sentiment)) +
  geom_bar(aes(fill = sentiment)) +
  scale_fill_brewer(palette = "RdGy") +
  labs(fill = "Sentiment") +
  labs(x = "Sentiment Categories", y = "Number of Comments") +
  ggtitle("Sentiment Analysis of Comments for Alejandro")


emo_alejandro_scores <- get_nrc_sentiment(clean_alejandro_text)[ , 1:8]
emo_alejandro_scores_df <- data.frame(clean_alejandro_text, emo_alejandro_scores)
emo_alejandro_sums <- emo_alejandro_scores_df[,2:9] |>
  sign() |>
  colSums() |>
  sort(decreasing = TRUE) |>
  data.frame() / nrow(emo_alejandro_scores_df)
names(emo_alejandro_sums)[1] <- "Proportion"

# Plot emotion classification
ggplot(emo_alejandro_sums, aes(x = reorder(rownames(emo_alejandro_sums), Proportion),
                               y = Proportion,
                               fill = rownames(emo_alejandro_sums))) +
  geom_col() +
  coord_flip()+
  guides(fill = "none") +
  scale_fill_brewer(palette = "Dark2") +
  labs(x = "Emotion Categories", y = "Proportion of Comments") +
  ggtitle("Emotion Analysis of Comments for Alejandro")
```

Figure 58 R code for sentiment analysis of Alejandro YouTube video

we begins by cleaning the text data, which involves removing URLs, HTML tags, non-ASCII characters, word elongations, internet slang, contractions, numbers, and punctuation. Next, the code assigns sentiment scores to the cleaned text using the AFINN lexicon. The sentiment scores are then stored in a data frame, and the scores are converted to categorical labels (Positive, Neutral, Negative). In addition to sentiment analysis, the code performs emotion analysis using the NRC emotion lexicon, which assigns scores for eight different emotions: anger, anticipation, disgust, fear, joy, sadness, surprise, and trust.

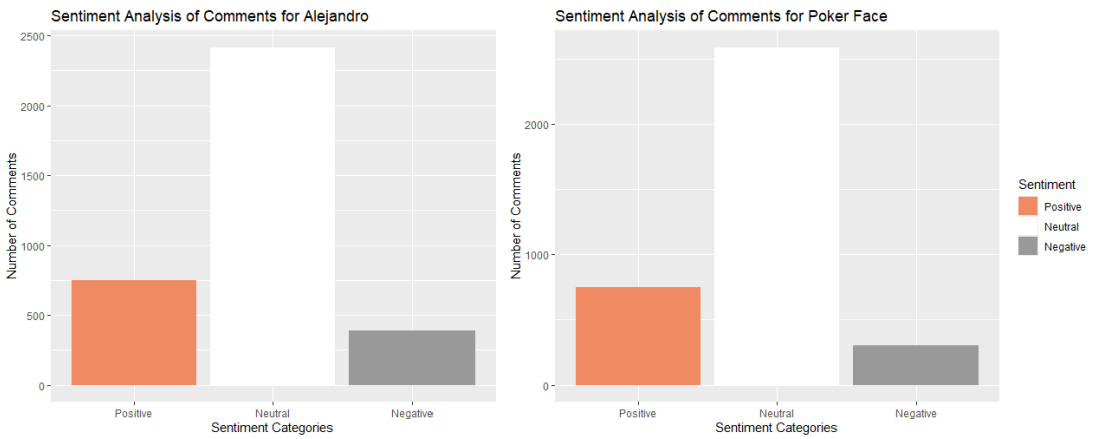


Figure 59 sentiment analysis results for the two videos, "Alejandro" and "Poker Face"

The sentiment analysis results for the two videos, "Alejandro" and "Poker Face," show interesting patterns in the audience's reactions. Both videos have a similar number of positive comments, around 750, but they differ in the number of negative comments, with "Alejandro" having around 400 negative comments and "Poker Face" having around 300.

Several factors might contribute to these differences:

Content and Theme:
"Alejandro" has a more complex and potentially controversial theme compared to "Poker Face." The song deals with themes of love, loss, and religious imagery, which might provoke stronger and more polarized reactions from the audience. This could explain the higher number of negative comments for "Alejandro."

Visual Style and Imagery:
The music video for "Alejandro" is known for its bold and provocative imagery, which includes military and religious symbols. Such imagery can be polarizing, attracting both strong approval and strong disapproval from viewers. On the other hand, "Poker Face" has a more straightforward and less controversial visual style, which might lead to fewer negative reactions.

Timing and Cultural Context:
The cultural context and timing of the release of these videos can also influence audience reactions. "Alejandro" was released after Lady Gaga had already established herself as a bold and avant-garde artist, which might lead to higher scrutiny and stronger opinions. "Poker Face," released earlier in her career, might have benefited from the novelty and fresh appeal of her style at the time.

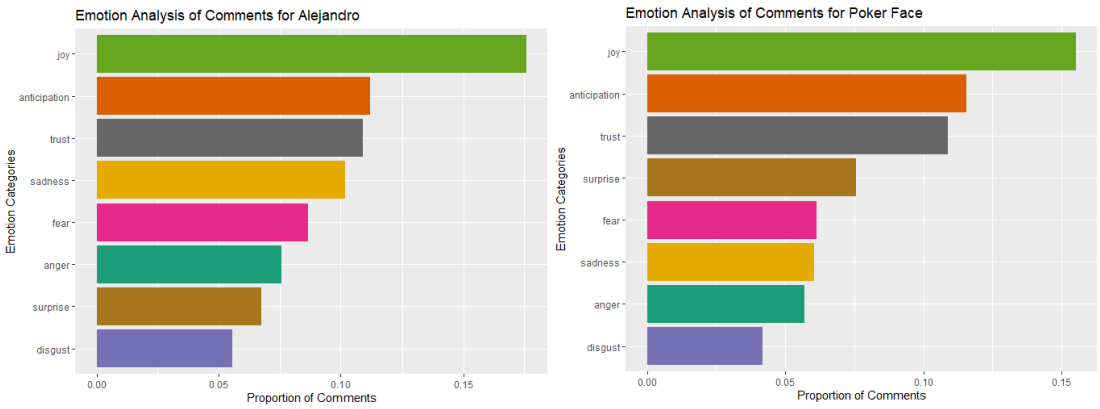


Figure 60 Emotion Analysis of Comments for "Alejandro" and "Poker Face"

The emotion analysis of comments for the two videos, "Alejandro" and "Poker Face," provides deeper insights into how the audience emotionally reacts to these music videos. Different with sentiment analysis by positive and negative bar chart, the bar charts above show the proportion of comments expressing detailly different emotions for each video.

Joy

Joy is the most common emotion, indicating that the video brings happiness to a large portion of its viewers. The "joy" score proportion is the highest for both videos, suggesting that the majority of the audience responds positively to the content. This aligns with the sentiment analysis, which shows a high number of positive comments for both videos, indicating an overall good emotional response and positive feedback towards the content.

Anticipation and Trust

Following "joy" are "anticipation" and "trust," which reflect the excitement and eagerness of the audience as well as their confidence in the content. These emotions are crucial positive indicators. Although the rankings of "anticipation" and "trust" are the same for both videos, the scores for Alejandro are noticeably lower than those for Poker Face. This suggests that while both videos generate excitement and trust, "Poker Face" does so more effectively. This could be due to the more upbeat and catchy nature of "Poker Face," which might resonate more strongly with the audience, generating higher anticipation and trust.

Sadness

The next prominent emotion is "sadness," which reflects the melancholic aspects of the songs. The sadness score for "Poker Face" is relatively low, around 0.06, indicating that the song does not evoke a strong negative emotional response. In contrast, "Alejandro" has a higher sadness score of 0.1. This suggests that the overall tone and

narrative of "Alejandro" are more likely to evoke feelings of sorrow or melancholy among viewers, which could lead to sad or unhappy feedback.

Negative Emotions: Fear, Anger, and Disgust

Alejandro has relatively higher proportions of negative emotions such as "fear," "anger," and "disgust" compared to "Poker Face." This indicates that "Alejandro" may contain elements that are more likely to provoke these negative reactions. The higher levels of fear and anger could be attributed to the darker themes and intense visuals of the video, which may evoke stronger negative feelings in some viewers. Disgust, though less common, still appears with a higher proportion in "Alejandro," suggesting that certain scenes or themes in the video might be unsettling for some viewers.

"Alejandro" evokes a broader range of emotions, including higher levels of sadness, fear, and anger. This emotional complexity suggests that the video may have more intricate themes and narratives, potentially leading to more varied audience reactions. In the other hand, the higher negative emotion scores for "Alejandro" indicate that its themes and visuals are more likely to evoke mixed reactions, potentially leading to more polarizing opinions. "Poker Face" has higher scores for anticipation and trust, reflecting a more straightforward and engaging presentation that fosters a stronger sense of excitement and confidence among viewers.

5.2 Decision Tree

```r
# Get songs from Lady Gaga and their audio features
LadyGaga_features <- get_artist_audio_features("Lady Gaga")
View(LadyGaga_features)

# Select relevant columns for the model
LadyGaga_features_subset <- LadyGaga_features[ , 9:20]
View(LadyGaga_features_subset)

# Get top 50 songs and their audio features
top50_features <- get_playlist_audio_features("spotify", "37i9dQZF1DXcBWIGoYBM5M")
View(top50_features)

# Select relevant columns for the model
top50_features_subset <- top50_features[ , 6:17]
top50_features_subset <- top50_features_subset |> rename(track_id = track.id)
View(top50_features_subset)

# Add the 'isLadyGaga' column (class variable) to each data frame
top50_features_subset["isLadyGaga"] <- 0
LadyGaga_features_subset["isLadyGaga"] <- 1

# Remove any songs by Lady Gaga that appear in the top 50
top50_features_noLadyGaga <- anti_join(top50_features_subset, LadyGaga_features_subset, by = "track_id")

# Combine the two data frames into one dataset
comb_data <- rbind(top50_features_noLadyGaga, LadyGaga_features_subset)

# Format the dataset
comb_data$isLadyGaga <- factor(comb_data$isLadyGaga)
comb_data <- select(comb_data, -track_id)

# Randomise the dataset (shuffle the rows)
comb_data <- comb_data[sample(1:nrow(comb_data)), ]

# Split the dataset into training and testing sets (80% training, 20% testing)
split_point <- as.integer(nrow(comb_data)*0.8)
training_set <- comb_data[1:split_point, ]
testing_set <- comb_data[(split_point + 1):nrow(comb_data), ]

# Train the decision tree model
dt_model <- train(isLadyGaga~ ., data = training_set, method = "C5.0")
```

Figure 61 R code for Decision Tree of Spotify Lady Gaga's data set (original)

Firstly, we collect the audio features of songs by Lady Gaga using the get_artist_audio_features function from the spotifyr package and then select the relevant columns (audio features and track ID) for the model. Then retrieves the audio features of the top 50 songs on Spotify and selects the relevant columns.

To differentiate between Lady Gaga's songs and others, a new column isLadyGaga is added to each dataset, with Lady Gaga's songs labeled as 1 and others as 0. we also removes any Lady Gaga songs that appear in the top 50 playlist to avoid duplication and combines the two datasets.

```r
# Sample a single prediction
prediction_row <- 1 # MUST be smaller than or equal to training set size

predicted_label <- predict(dt_model, testing_set[prediction_row, ])
predicted_label <- as.numeric(levels(predicted_label))[predicted_label]

if (predicted_label == testing_set[prediction_row, 12]){
  print(paste0("Prediction is: ", predicted_label, ". Correct!"))
} else {
  print(paste0("Prediction is: ", predicted_label, ". Wrong."))
}

# Analyse the model accuracy with a confusion matrix
confusionMatrix(predict(dt_model, testing_set), reference = testing_set$isLadyGaga)
```

```
+ }
[1] "Prediction is: 1. Wrong."
> # Analyse the model accuracy with a confusion matrix
> confusionMatrix(predict(dt_model, testing_set), reference = testing_set$isLadyGaga)
```

```
          Reference
Prediction  0  1
         0  2  2
         1  7 50

               Accuracy : 0.8525
                 95% CI : (0.7383, 0.9302)
    No Information Rate : 0.8525
    P-Value [Acc > NIR] : 0.5877

                  Kappa : 0.2386

 Mcnemar's Test P-Value : 0.1824

            Sensitivity : 0.22222
            Specificity : 0.96154
         Pos Pred Value : 0.50000
         Neg Pred Value : 0.87719
             Prevalence : 0.14754
         Detection Rate : 0.03279
   Detection Prevalence : 0.06557
      Balanced Accuracy : 0.59188

       'Positive' Class : 0
```

Figure 62 Testing and Prediction result

The confusion matrix provides an accuracy rate of 85.25%, yet we observe instances of wrong predictions. This discrepancy is primarily due to the imbalanced dataset, where we have a substantial amount of data on Lady Gaga's songs but relatively fewer songs from other artists. As a result, the model excels at predicting songs with features similar to Lady Gaga's, demonstrating high accuracy in these cases. However, it shows lower accuracy when predicting songs that are not by Lady Gaga. To improve the model's performance, especially in identifying non-Lady Gaga songs, we should increase the dataset size for other artists' songs. For instance, instead of using the top 50 songs, we could expand the dataset to include the more songs from various artists, such as "Beyoncé", "Drake" and "Taylor Swift". This approach would provide a more balanced dataset, enhancing the model's ability to accurately classify both Lady Gaga's and other artists' songs.

```
# List of other artists to include in the dataset
other_artists <- c( "Beyoncé", "Drake", "Taylor Swift")

# Get audio features for the other artists
other_artists_features <- get_other_artists_audio_features(other_artists)

# Combine Lady Gaga and other artists data
comb_data <- rbind(LadyGaga_features_subset, other_artists_features)

# Format the dataset
comb_data$isLadyGaga <- factor(comb_data$isLadyGaga)
comb_data <- select(comb_data, -track_id)

# Randomize the dataset (shuffle the rows)
set.seed(42)  # For reproducibility
comb_data <- comb_data[sample(1:nrow(comb_data)), ]

# Split the dataset into training and testing sets (80% training, 20% testing)
split_point <- as.integer(nrow(comb_data) * 0.8)
training_set <- comb_data[1:split_point, ]
testing_set <- comb_data[(split_point + 1):nrow(comb_data), ]

# Train the decision tree model
dt_model <- train(isLadyGaga ~ ., data = training_set, method = "C5.0")
```

```
+ }
[1] "Prediction is: 0. Correct!"
```

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 232  23
         1   7  30

               Accuracy : 0.8973
                 95% CI : (0.8566, 0.9296)
    No Information Rate : 0.8185
    P-Value [Acc > NIR] : 0.000139

                  Kappa : 0.6082

 Mcnemar's Test P-Value : 0.006170

            Sensitivity : 0.9707
            Specificity : 0.5660
         Pos Pred Value : 0.9098
         Neg Pred Value : 0.8108
             Prevalence : 0.8185
         Detection Rate : 0.7945
   Detection Prevalence : 0.8733
      Balanced Accuracy : 0.7684

       'Positive' Class : 0
```

Figure 63 Prediction result after model improvement

The improved decision tree model demonstrates a significant enhancement in performance compared to the previous version. The accuracy of the improved model increased to 89.73% from 85.25%, indicating more reliable overall performance. Additionally, the confidence interval is now narrower, suggesting a more precise accuracy estimation. The P-Value improved drastically, indicating that the model's accuracy is statistically significantly better than the no-information rate. The Kappa statistic, which measures the agreement between predicted and actual classifications, improved from 0.2386 to 0.6082, showing a substantial increase in the model's reliability. Furthermore, the sensitivity of the improved model increased dramatically from 22.22% to 97.07%, highlighting its effectiveness at correctly identifying Lady Gaga's songs.

5.3 Topic Modelling

```r
# Clean the text data for Alejandro
alejandro_clean_text <- alejandro_yt_data$Comment |>
  replace_url() |>
  replace_html() |>
  replace_non_ascii() |>
  replace_word_elongation() |>
  replace_internet_slang() |>
  replace_contraction() |>
  removeNumbers() |>
  removePunctuation()

# Convert cleaned text to a document corpus
alejandro_text_corpus <- VCorpus(VectorSource(alejandro_clean_text))

# Example of viewing the content of specific documents
alejandro_text_corpus[[1]]$content
alejandro_text_corpus[[5]]$content

# Further preprocessing: lowercasing, removing stopwords, and stripping whitespace
alejandro_text_corpus <- alejandro_text_corpus |>
  tm_map(content_transformer(tolower)) |>
  tm_map(removeWords, stopwords(kind = "SMART")) |>
  # tm_map(stemDocument) |>  # Optional stemming
  tm_map(stripWhitespace)

# Example of viewing the preprocessed content of specific documents
alejandro_text_corpus[[1]]$content
alejandro_text_corpus[[5]]$content

# Convert corpus to a Document Term Matrix and remove zero entries
alejandro_doc_term_matrix <- DocumentTermMatrix(alejandro_text_corpus)
alejandro_non_zero_entries <- unique(alejandro_doc_term_matrix$i)
alejandro_dtm <- alejandro_doc_term_matrix[alejandro_non_zero_entries, ]


# Create LDA model with k topics for Alejandro
alejandro_lda_model <- LDA(alejandro_dtm, k = 6)  # Adjust k as needed

# Generate topic probabilities for each word ('beta')
alejandro_found_topics <- tidy(alejandro_lda_model, matrix = "beta")
View(alejandro_found_topics)
```

Figure 64 R code for topic modelling

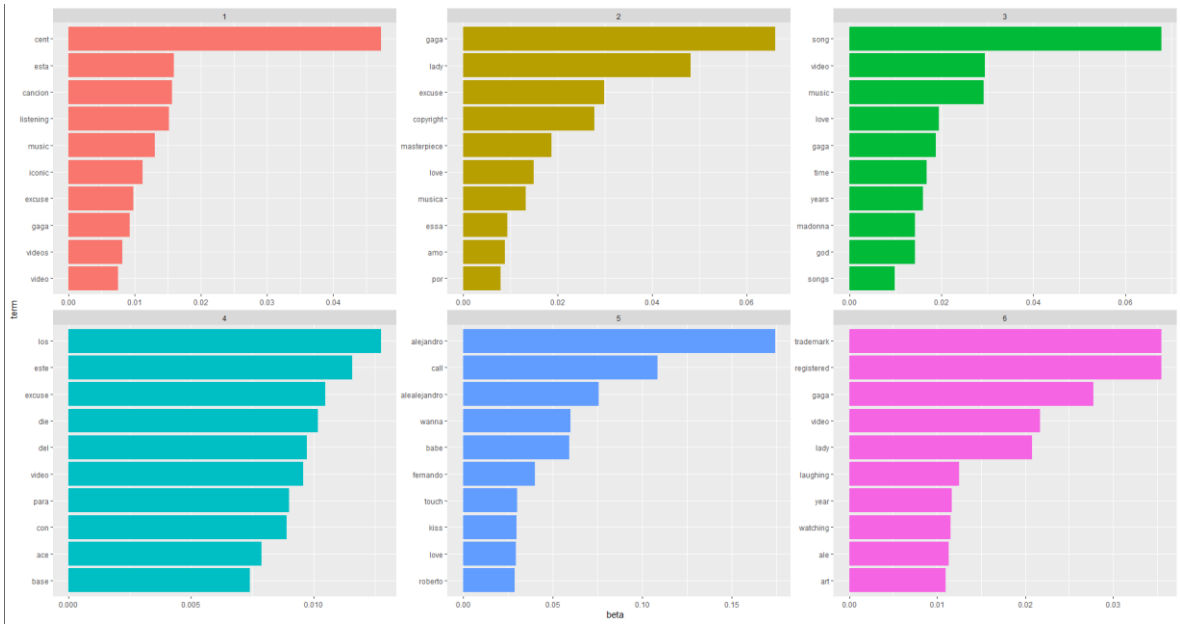


Figure 65 Topic Modeling Result of Alejandro

The results from the LDA topic modeling provide insights into the main themes discussed in the comments of the Alejandro YouTube video. we can get different

analysis of each topic based on the top terms, for example:

From the graph, the Top terms in topic 5 including trademark, registered, gaga, video, lady, laughing, year, watching, ale, art. This top topic is directly related to the lyrics and narrative of the song "Alejandro." Terms like "alejandro," "call," "alealejandro," and "wanna" are all parts of the song's lyrics. The names "Fernando" and "Roberto" also appear in the lyrics, suggesting discussions about the song's content and story.

In topic 2, the top terms including gaga, lady, excuse, copyright, masterpiece, love, musica, essa, amo, por. This topic is centered on Lady Gaga herself, as indicated by the prominent terms "gaga" and "lady." Words like "masterpiece" and "love" suggest that viewers are expressing admiration for the song and Lady Gaga's artistry. "Copyright" might indicate discussions around the ownership or use of the song.

The results indicate that fans are highly engaged with Lady Gaga's music, as evidenced by the frequent use of terms like "iconic," "masterpiece," "love," and "music." This suggests a strong admiration and positive reception of Lady Gaga's work. The mention of other artists, particularly "Madonna," indicates that viewers are drawing comparisons between Lady Gaga and other well-known music icons. This suggests that Lady Gaga is often viewed in the context of her influence and legacy within the pop music industry. Such comparisons can spark discussions about her artistic style, impact, and place in the history of pop music.

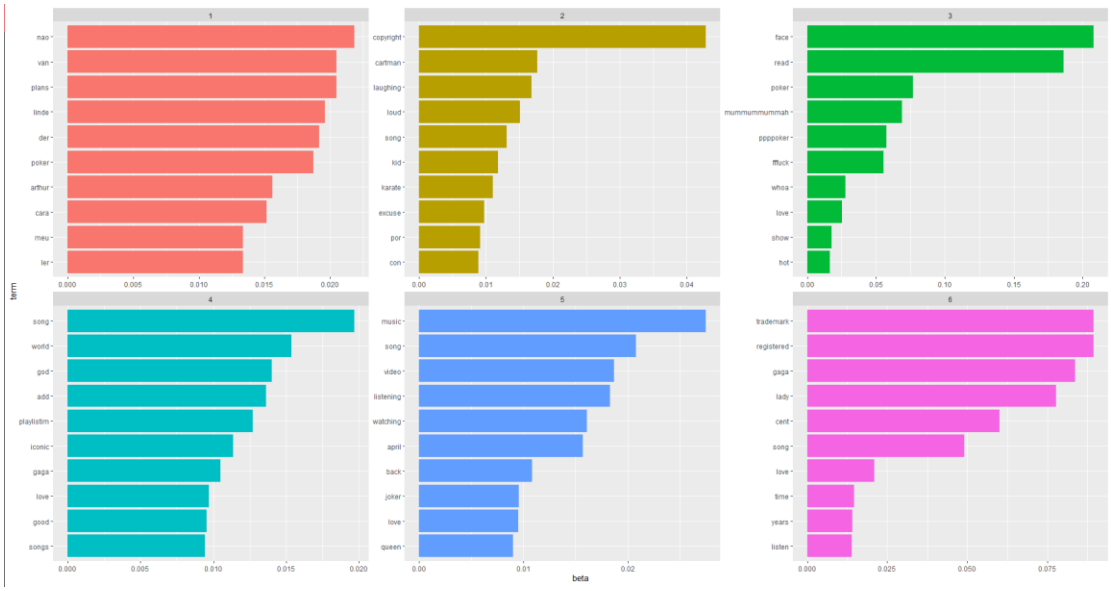


Figure 66 Topic Modeling Result of Poker Face

The topic modeling results for "Poker Face" show different prominent themes compared to "Alejandro." The most significant terms include "face," "song," "music,"

and "video," indicating that fans are heavily focused on the song's title and its overall musical and video presentation. There are frequent mentions of "trademark" and "registered," suggesting discussions about copyright or the song's iconic status. The appearance of terms like "karaoke," "laughing," and "loud" suggests a more playful and entertaining engagement with "Poker Face," reflecting its upbeat and catchy nature. Fans also discuss the song's performance aspects, similar to "Alejandro," but with a focus on different elements like "hot," "show," and "mummummummah," which are specific to the "Poker Face" lyrics and presentation.

Both "Alejandro" and "Poker Face" have discussions centered around Lady Gaga and the songs themselves, indicating strong fan engagement and appreciation. However, "Alejandro" has a more emotionally charged discussion with terms like "love," "babe," and "excuse," reflecting its deeper emotional and narrative themes. "Poker Face," on the other hand, shows a playful and entertaining theme with terms like "karaoke," "laughing," and "loud," indicating that fans engage with this song in a more lighthearted manner.

"Alejandro" has terms that highlight its narrative and emotional impact, such as "call," "excuse," and "babe," suggesting that fans are deeply engaged with the song's storytelling. "Poker Face" discussions are more focused on its catchy lyrics and presentation, with terms like "mummummummah" and "face," indicating fans' enjoyment of the song's memorable and playful lyrics.

The topic modeling results provide valuable insights into how fans engage with "Alejandro" and "Poker Face." While both songs generate significant fan engagement and appreciation, "Alejandro" elicits more emotional and narrative-driven discussions, whereas "Poker Face" inspires playful and entertaining interactions.

6. Dashboard

6.1 Stacked bar chart: Count of Comment by Name

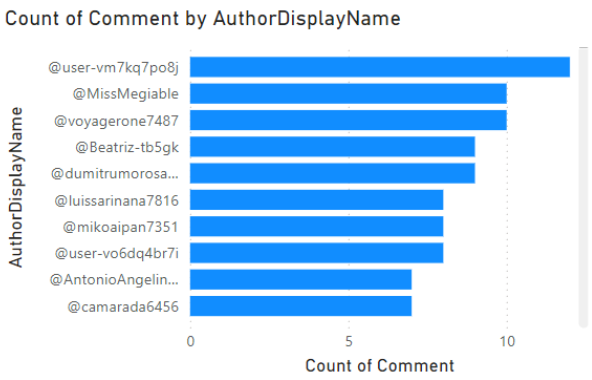


Figure 67 Count of Comment by AuthorDisplayName

This bar chart shows the number of comments made by the top commenters on the "Alejandro" video. Each bar represents a different user, with the length of the bar corresponding to the number of comments made.

This chart helps identify the most active users in the comment section. For example, @user-xm7kzp0oj8 has made the most comments, followed by @MissMegaBite and @voyageorin7487. Identifying top commenters can help in recognizing influential members of the community. These users might be driving discussions and influencing other viewers' engagement. This data can be used to target specific users for community-building efforts, such as shout-outs, direct interactions, or incentives to further boost their engagement.

6.2 Card: Total Counts

Figure 68 Total Number of Comments and Reply Comments

The total of 3347 comments and 1355 replies provides a quick snapshot of overall engagement. This high-level metric sets the context for more detailed analysis provided by the other charts.

6.3 Pie Chart: Count of Comment by Month

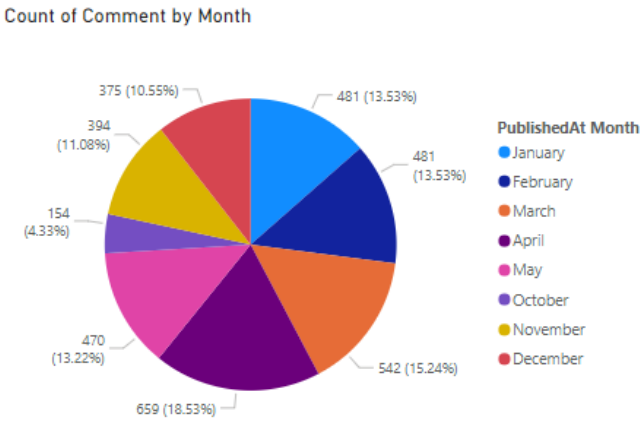


Figure 69 Count of Comment by Month

This pie chart displays the distribution of comments across different months. Each segment represents a month, showing the percentage of total comments made in that month.

The chart reveals that October has the highest comment count (18.53%), followed by May (15.24%) and February (13.53%). These peaks might correlate with events related to Lady Gaga, such as new releases, anniversaries, or social media activities. Understanding when engagement is highest can inform future content releases. For instance, planning major content drops or marketing campaigns around months with historically high engagement can capitalize on this trend.

6.4 Word Cloud

Figure 70 Alejandro Word Cloud

The word cloud visualizes the most frequently used words in the comments. Larger words indicate higher frequency.

This chart provides a quick overview of the main topics and emotional tone of the discussion, which can be valuable for content analysis and understanding audience sentiment. The prevalence of words like "love" indicates a generally positive sentiment among the comments. This can help gauge the overall reception of the video.

6.5 Tree Map: Name of Reply Comment

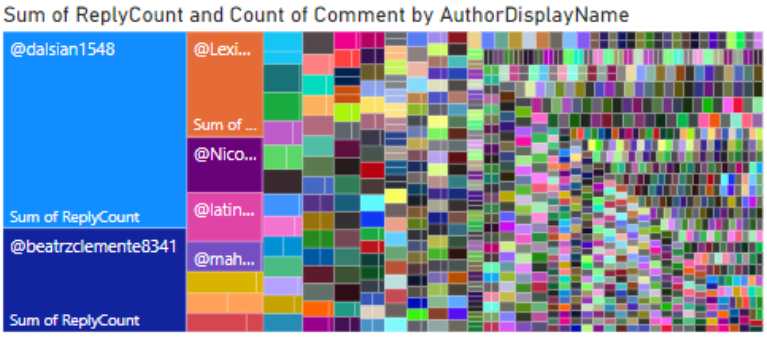


Figure 71 Sum of ReplyCount and Count of Comment by AuthorDisplayName

This matrix/trees shows the sum of reply counts and the count of comments for various users. It combines the number of comments with the number of replies each user received.

This chart offers a detailed view of user interaction dynamics. For example, @dalisan1548 and @beatrclemente8341 have high reply counts, indicating that their comments generate more discussion. This can help identify influential users and understand which comments drive engagement. This matrix/trees also provides a nuanced view of engagement by combining both comments and replies, offering a deeper understanding of interaction patterns among users, by combining this matrix with the above chart we can easily analysis the number of reply comments and who been reply in ever moth.

6.6 Combined Dashboard

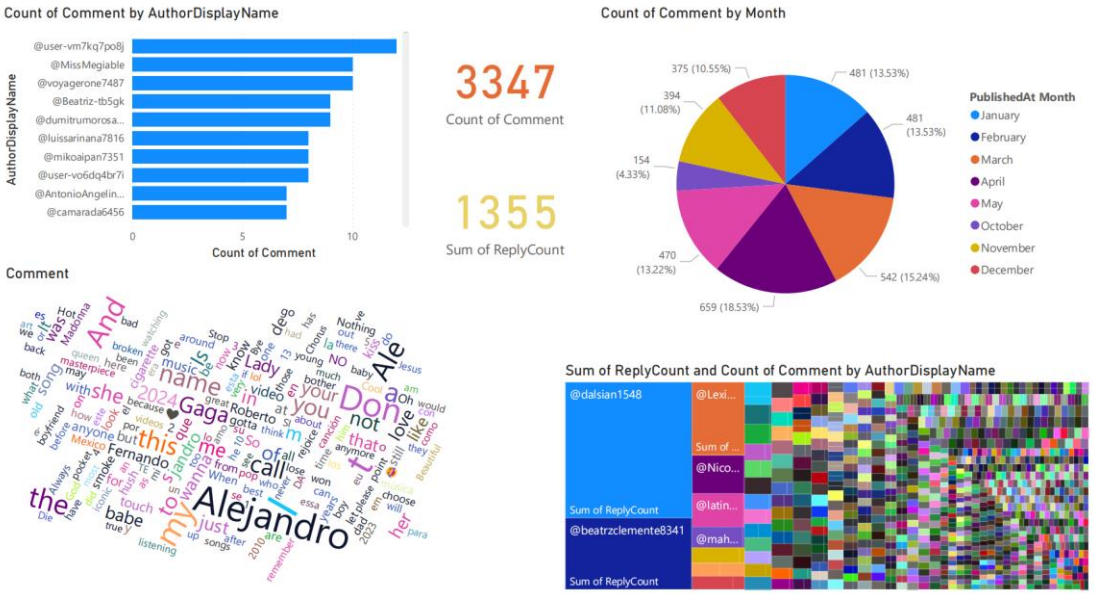


Figure 72 Alejandro Dashboard

The Power BI dashboard for analyzing comments on Lady Gaga's "Alejandro" video provides a comprehensive and interactive tool for understanding user engagement and interaction patterns. By combining five different charts, this dashboard offers a detailed view of various aspects of the data, allowing for deep insights through interactive exploration.

The interactive elements and cross-filtering capabilities of the dashboard are particularly noteworthy. When these charts are combined in a single dashboard, they enable cross-filtering and interactive analysis, significantly enhancing the insights one can draw from the data.

When interacting with the pie chart, for example by clicking on the segment representing February, the entire dashboard updates to reflect data specific to that month. The word cloud will update to display the most frequently used words in comments made in February. This allows for a quick sentiment and topic analysis specific to that month, revealing what themes or issues were most prominent during that period. The bar chart showing the count of comments by author will filter to show only those users who commented in February. This highlights the most active users for that particular month, providing insights into who was driving engagement during this time. Additionally, the matrix showing the sum of reply counts and count of comments by author will highlight users who not only commented but also received replies in February. This shows the influence and interaction dynamics of specific users during that month. The total count of comments and replies will update to reflect only those

made in February, providing a quick snapshot of engagement levels for that specific month. By clicking on different months in the pie chart, one can quickly compare engagement patterns across different time periods, identifying trends and seasonal variations.

When interacting with the bar chart by clicking on a specific username, the pie chart will update to show the distribution of comments by month for the selected user. This reveals when this user was most active and can help correlate their activity with specific events or releases. The word cloud will display the most frequently used words by the selected user, providing insights into the topics and sentiments expressed by that user, showing what aspects of the video or related topics they focus on. The matrix chart will highlight the replies received by the selected user, showing their influence and how others interact with them. The total count will update to reflect the number of comments and replies made by the selected user, providing a clear view of their overall engagement. By clicking on different usernames, one can compare the engagement patterns, topics, and influence of different users, identifying key influencers and active participants.

The combined interactive functionality of the dashboard allows for several in-depth analyses. By filtering by month or user, one can identify patterns in user engagement, such as peak activity periods, influential users, and common topics. This helps understand how engagement evolves over time and what factors drive it. The word cloud, when filtered by month or user, provides a quick way to analyze the sentiment and topics discussed. This can reveal how the audience's perception changes over time or what specific users focus on. The matrix chart showing replies helps identify key influencers within the community. Users who receive many replies are likely driving conversations and shaping the community's interaction patterns. The ability to compare different time periods or users side by side helps in understanding the dynamics of engagement and interaction. This can reveal trends, seasonal effects, and the impact of specific events or releases on user activity.

7. Analysis Review

In our analysis of Lady Gaga's YouTube videos "Alejandro" and "Poker Face," we employed methods such as sentiment analysis, topic modeling, and decision trees. While these methods provided valuable insights, exploring other advanced algorithms and techniques could potentially enhance our analysis and yield deeper insights. I review and compare alternative methods and algorithms, such as Random Forests and Gradient Boosting Machines (GBM), which could significantly enhance performance.

For instance, our initial use of decision trees yielded an accuracy rate of around 85.25%, which, while decent, highlighted the model's limitations, especially with imbalanced datasets. The decision tree model struggled with lower accuracy in predicting non-Lady Gaga songs due to the smaller dataset of other artists. This was evident in our confusion matrix analysis, which showed a high accuracy rate for Lady Gaga songs but a much lower rate for others.

Random Forests: as described in the referenced article by Varun Samarth (Samarth, 2023), create an ensemble of decision trees and average their predictions to reduce variance and overfitting. This method provides robust predictive accuracy even with noisy data and can handle large datasets with many features. Random Forests can be used for both regression and classification tasks, making them versatile for our analysis. The random selection process in Random Forests helps in capturing complex interactions between variables, which might lead to more accurate predictions compared to a single decision tree. For example, when we improved our dataset by including more songs from other artists, the Random Forest model could better distinguish between Lady Gaga and non-Lady Gaga songs, significantly enhancing predictive accuracy.

Gradient Boosting Machines: GBM, including its popular variant XGBoost, build trees sequentially, with each tree correcting the errors of the previous ones. This method often results in higher accuracy and better handling of complex relationships in the data. GBM can provide significant improvements in predictive power and model robustness, making it a strong candidate for enhancing our classification of Lady Gaga's songs. As highlighted in the Gradient Boosting article (Snowflake, 2023), this technique can handle large datasets efficiently and improve prediction accuracy, especially in noisy environments. GBM's iterative approach ensures that each new tree improves on the mistakes of the previous ones, leading to a model that is not only more accurate but also more resilient to overfitting. This would have been particularly useful in our case where the decision tree model showed limitations due to overfitting issues.

Overall, integrating these advanced machine learning models such as Random Forests and Gradient Boosting Machines can provide a more robust and accurate analysis. They can handle complex datasets better and offer improved prediction capabilities, thereby providing deeper insights into the data. This approach can significantly enhance our ability to analyze and predict user behavior and preferences, leading to more informed decisions to boost Lady Gaga's popularity and fan engagement.

8. Conclusion and Suggestions

This report aims to provide a comprehensive analysis of Lady Gaga's YouTube videos "Alejandro" and "Poker Face," along with insights from Reddit and Spotify data. The objective is to identify strategies to enhance Lady Gaga's visibility and fan base by leveraging data analytics.

For data collection, we extracted 3,000 comments each from the YouTube videos "Alejandro" and "Poker Face" to analyze public reception. Additionally, a Reddit thread discussing Lady Gaga's iconic meat dress was selected to understand public discourse on a controversial topic. Spotify data was also retrieved to analyze her music's audio features and collaboration patterns. Actor networks for "Alejandro" and "Poker Face" revealed differences in user engagement, with the top commenter for "Alejandro" having a significantly higher PageRank score, suggesting more intense and possibly polarizing discussions.Frequent words and PageRank bigrams highlighted key discussion topics and influential terms within the comments. For instance, frequent words in the "Poker Face" video included "face," "poker," and "mumummummmah," reflecting engagement with the song's lyrics and themes. In contrast, "Alejandro" discussions centered around terms like "alejandro," "call," and "alealejandro," indicating engagement with the narrative and storytelling elements. Comparing the results, "Alejandro" elicited more emotionally charged discussions, while "Poker Face" inspired playful and entertaining interactions.

Sentiment analysis showed that both videos had a similar number of positive comments, around 750 each. However, "Alejandro" had more negative comments (around 400) compared to "Poker Face" (around 300). This difference suggests that while both videos are well-received, "Alejandro" elicits more polarized reactions. Emotion analysis indicated that "Alejandro" had higher proportions of sadness, fear, anger, and disgust compared to "Poker Face," reflecting its complex emotional themes and provocative visuals.

From the analysis of Spotify data, we observed that Lady Gaga's more lively and upbeat songs are generally more popular, which aligns with why "Poker Face" is more well-received compared to "Alejandro." Reddit discussions about Lady Gaga, particularly focusing on her iconic meat dress, revealed a mix of admiration, curiosity, and controversy. The analysis showed that while there is substantial support and interest in her bold fashion choices, there are also critical voices that contribute to a balanced discourse. This duality in public perception is crucial for understanding the broader impact of her persona and public image.

To enhance Lady Gaga's popularity and fan base, it is recommended to leverage

emotionally charged content like "Alejandro" to deepen fan connections, promote upbeat songs like "Poker Face" to attract a broader audience, engage with key influencers to amplify discussions, balance content themes to cater to diverse audience preferences, and continuously update and expand the dataset for accurate analysis.

References

billboard. (2023, March 19). *Lady Gaga*. Retrieved from billboard: https://www.billboard.com/artist/lady-gaga/

Denise Winterman, Jon Kelly. (2010, September 14). *Five interpretations of Lady Gaga's meat dress*. Retrieved from BBC News Magazine: https://www.bbc.com/news/magazine-11297832

Godga. (2023, Sepetember 15). *Lady Gaga Albums Ranked*. Retrieved from albumoftheyear: https://www.albumoftheyear.org/user/notgod/list/154679/lady-gaga-albums-ranked/

Levy, M. (2024, May 19). *Lady Gaga*. Retrieved from britannica: https://www.britannica.com/biography/Lady-Gaga

Montalti, V. (2022, March 29). *15 surprising things you might not know about Lady Gaga*. Retrieved from businessinsider: https://www.businessinsider.com/surprising-facts-about-lady-gaga#lady-gagas-debut-acting-credit-was-in-an-episode-of-the-sopranos-at-age-15-4

Samarth, V. (2023, December 14). *What is Random Forest In Data Science and How Does it Work?* Retrieved from emeritus: https://emeritus.org/in/learn/data-science-random-forest/#:~:text=Random%20forest%20is%20an%20ensemble,with%20random%20subsets%20of%20data

Snowflake. (2023, March 20). *WHAT IS GRADIENT BOOSTING?* Retrieved from snowflake: https://www.snowflake.com/guides/what-gradient-boosting/#:~:text=Gradient%20boosting%20is%20an%20ensemble,boosted%20decision%20trees%20(GBDTs)

Wong, C. M. (2024, March 12). *Lady Gaga Posts Fierce Defense Of Dylan Mulvaney Following Anti-LGBTQ+ Hatred*. Retrieved from yahoo: https://au.lifestyle.yahoo.com/lady-gaga-posts-fierce-defense-215746079.html