Machine Learning Social Media Analysis with R

Developed a data science workflow integrating data collection, network analysis, predictive modeling, and dashboard visualization to improve a music artist's social media engagement and popularity.

[DATE]

[COMPANY NAME]

[Company address]

Contents

1. Case Study Setting: Meghan Trainor 3

2. Data Selection & Exploration 3

2.1 Meghan Trainor YouTube music video 3

2.2 Pagerank top 5 most influential actor networks 3

2.3 Unique Actors 7

2.4 Use the Spotify API to extract data 8

2.4 Revalent features/ valence of Meghan Trainor’s songs 9

3. Text Pre-Processing........ 10

3.1 Reddit Posts Term-Document Matrices 10

3.2 Top10 highest frequency terms 11

3.2 Page rank semantic (bigram) networks 13

4. Social Network Analysis 15

4.1 Centrality Analysis: Degree, Betweenness, and Closeness 15

4.2 Community analysis with Girvan-Newman / Louvain methods ..20

5. Machine Learning Models 27

5.1 Sentiment Analysis on Public Reactions........ 27

5.2 Decision Tree Performance Prediction 32

5.3 LDA Topic Modelling to Identify Related Terms 34

6. Power BI Visualisation........ 35

6.1 YouTube Dashboard 36

6.2 Reddit Dashboard: 37

6.2 Spotify Dashboard: 38

7. Analysis Review 39

7.1 Eigenvector Centrality Network Analysis 40

8. Reference 41

1. Case Study Setting: Meghan Trainor

1) Meghan Trainor is an American singer-songwriter who has been active in the music industry since 2014, when she gained widespread recognition with her debut single "All About That Bass." Known for her retro-pop sound and empowering lyrics, Trainor has released five studio albums as of 2023, including Title (2015), Thank You (2016), and Takin' It Back (2022). She has published over 100 songs, many of which feature her signature blend of doo-wop, pop, and R&B influences. Trainor's music often focuses on themes of self-empowerment, body positivity, and relationships (Wikipedia, 2024).

2. Data Selection & Exploration

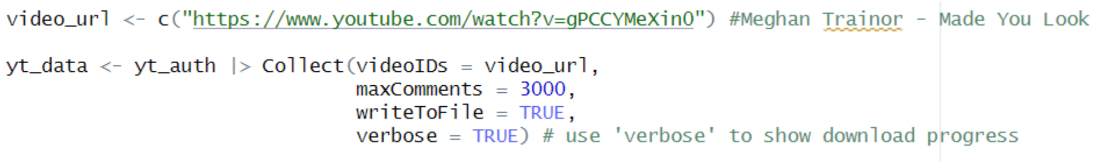
2.1 Meghan Trainor YouTube music video

Artist: Meghan Trainor from YouTube, song: Made You Look.

Meghan Trainor - Made You Look (Official Music Video)

I chose Maghan trainor because I love her song and Made You Look is one of her popular songs.

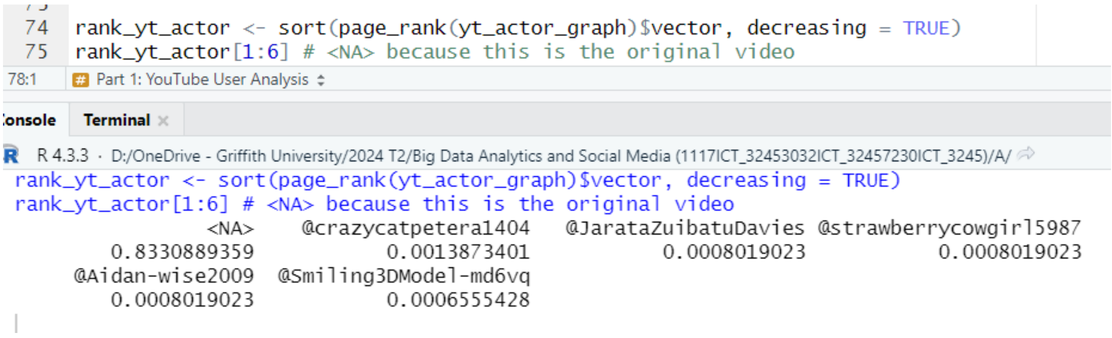
I collected 3000 comments under her video on Youtube.



```
video_url <- c("https://www.youtube.com/watch?v=gPCCYMeXin0") #Meghan Trainor - Made You Look

yt_data <- yt_auth |> Collect(videoIDs = video_url,
                              maxComments = 3000,
                              writeToFile = TRUE,
                              verbose = TRUE) # use 'verbose' to show download progress
```

2.2 Pagerank top 5 most influential actor networks

Create actor networks from your data and list the top 5 most influential actors for your artist/band according to page rank. Explain the results.



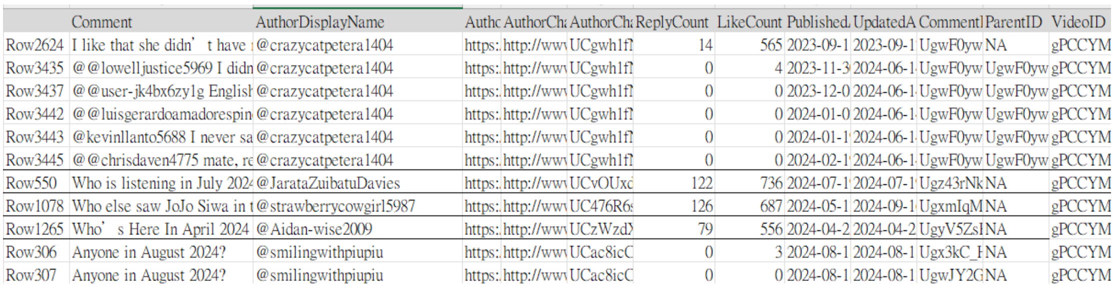
```
rank_yt_actor <- sort(page_rank(yt_actor_graph)$vector, decreasing = TRUE)
rank_yt_actor[1:6] # <NA> because this is the original video
```

78:1
Part 1: YouTube User Analysis
Console
Terminal
R 4.3.3 · D:/OneDrive - Griffith University/2024 T2/Big Data Analytics and Social Media (1117ICT_32453032ICT_32457230ICT_3245)/A/

```
rank_yt_actor <- sort(page_rank(yt_actor_graph)$vector, decreasing = TRUE)
rank_yt_actor[1:6] # <NA> because this is the original video
               <NA>    @crazycatpetera1404   @JarataZuibatuDavies @strawberrycowgirl5987 
       0.8330889359           0.0013873401           0.0008019023           0.0008019023 
    @Aidan-wise2009  @Smiling3DModel-md6vq 
       0.0008019023           0.0006555428 
```

@crazycatpetera1404, @JarataZuibatuDavies, @strawberrycowgirl5987, @Aidan-wise2009, @Smiling3DModel-md6vq

Below is the table of the top 5 PageRank score actors’ comment details.



	Comment	AuthorDisplayName	Authc	AuthorCh	AuthorCh	ReplyCount	LikeCount	Published	UpdatedA	Comment	ParentID	VideoID
Row2624	I like that she didn' t have	@crazycatpetera1404	https:	http://ww	UCgwh1f	14	565	2023-09-1	2023-09-1	UgwF0yw	NA	gPCCYM
Row3435	@@lowelljustice5969 I didn	@crazycatpetera1404	https:	http://ww	UCgwh1f	0	4	2023-11-3	2024-06-1	UgwF0yw	UgwF0yw	gPCCYM
Row3437	@@user-jk4bx6zy1g English	@crazycatpetera1404	https:	http://ww	UCgwh1f	0	0	2023-12-0	2024-06-1	UgwF0yw	UgwF0yw	gPCCYM
Row3442	@@luisgerardoamadorespin	@crazycatpetera1404	https:	http://ww	UCgwh1f	0	0	2024-01-0	2024-06-1	UgwF0yw	UgwF0yw	gPCCYM
Row3443	@kevinllanto5688 I never sa	@crazycatpetera1404	https:	http://ww	UCgwh1f	0	0	2024-01-1	2024-06-1	UgwF0yw	UgwF0yw	gPCCYM
Row3445	@@chrisdaven4775 mate, re	@crazycatpetera1404	https:	http://ww	UCgwh1f	0	0	2024-02-1	2024-06-1	UgwF0yw	UgwF0yw	gPCCYM
Row550	Who is listening in July 202	@JarataZuibatuDavies	https:	http://ww	UCvOUxd	122	736	2024-07-1	2024-07-1	Ugz43rNk	NA	gPCCYM
Row1078	Who else saw JoJo Siwa in t	@strawberrycowgirl5987	https:	http://ww	UC476R6	126	687	2024-05-1	2024-09-1	UgxmIqM	NA	gPCCYM
Row1265	Who' s Here In April 2024	@Aidan-wise2009	https:	http://ww	UCzWzd	79	556	2024-04-2	2024-04-2	UgyV5Zs	NA	gPCCYM
Row306	Anyone in August 2024?	@smilingwithpiupiu	https:	http://ww	UCac8icC	0	3	2024-08-1	2024-08-1	Ugx3kC_	NA	gPCCYM
Row307	Anyone in August 2024?	@smilingwithpiupiu	https:	http://ww	UCac8icC	0	0	2024-08-1	2024-08-1	UgwJY2G	NA	gPCCYM

In my analysis, I created an actor network from YouTube comments, where each actor represents a user, and the connections (edges) represent interactions like replies. Using the PageRank algorithm, I identified the top 5 most influential users:

@crazycatpetera1404 – 0.00138734

@JarataZuibatuDavies – 0.00080190

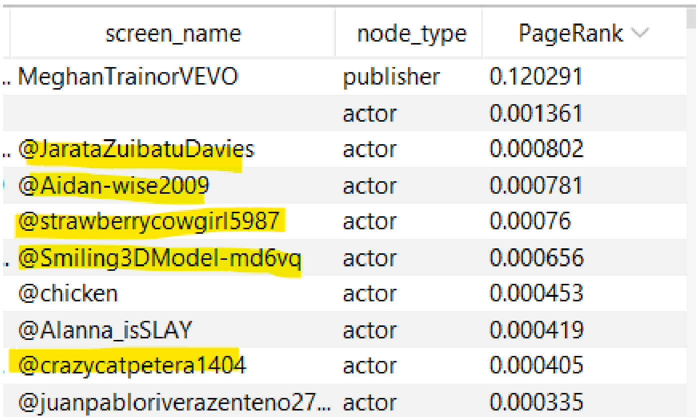
@strawberrycowgirl5987 – 0.00080190

@Aidan-wise2009 – 0.00065554

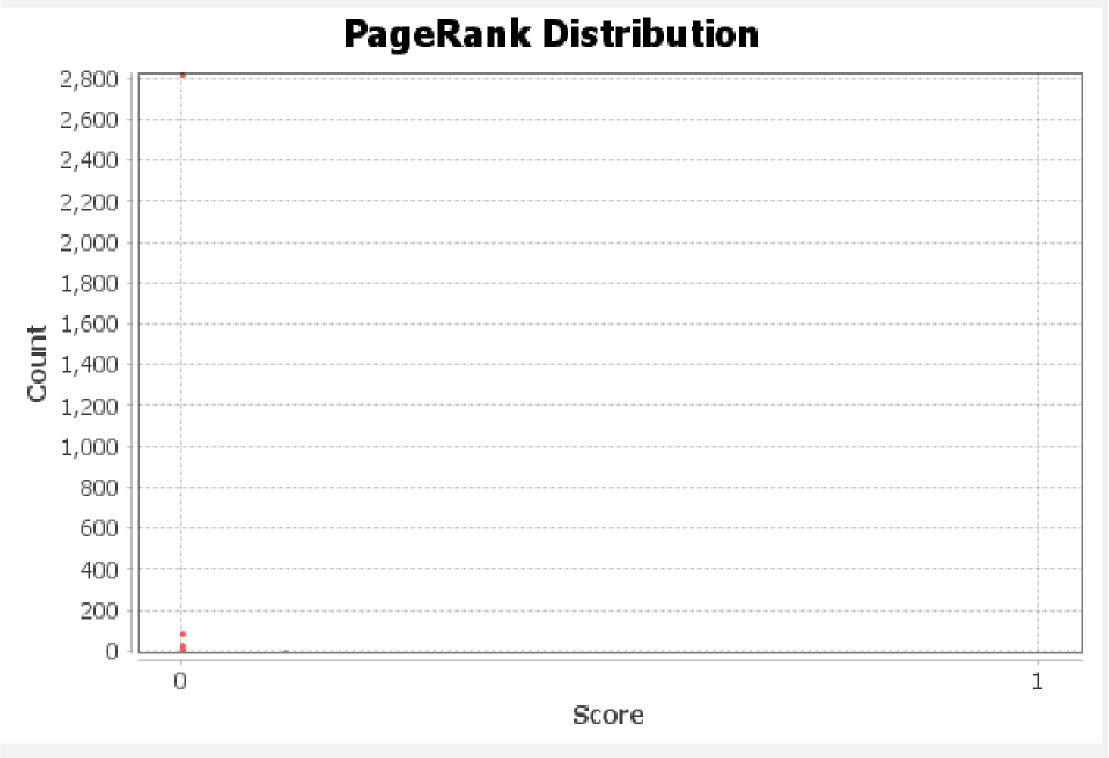
@Smiling3DModel-md6vq – 0.00065554

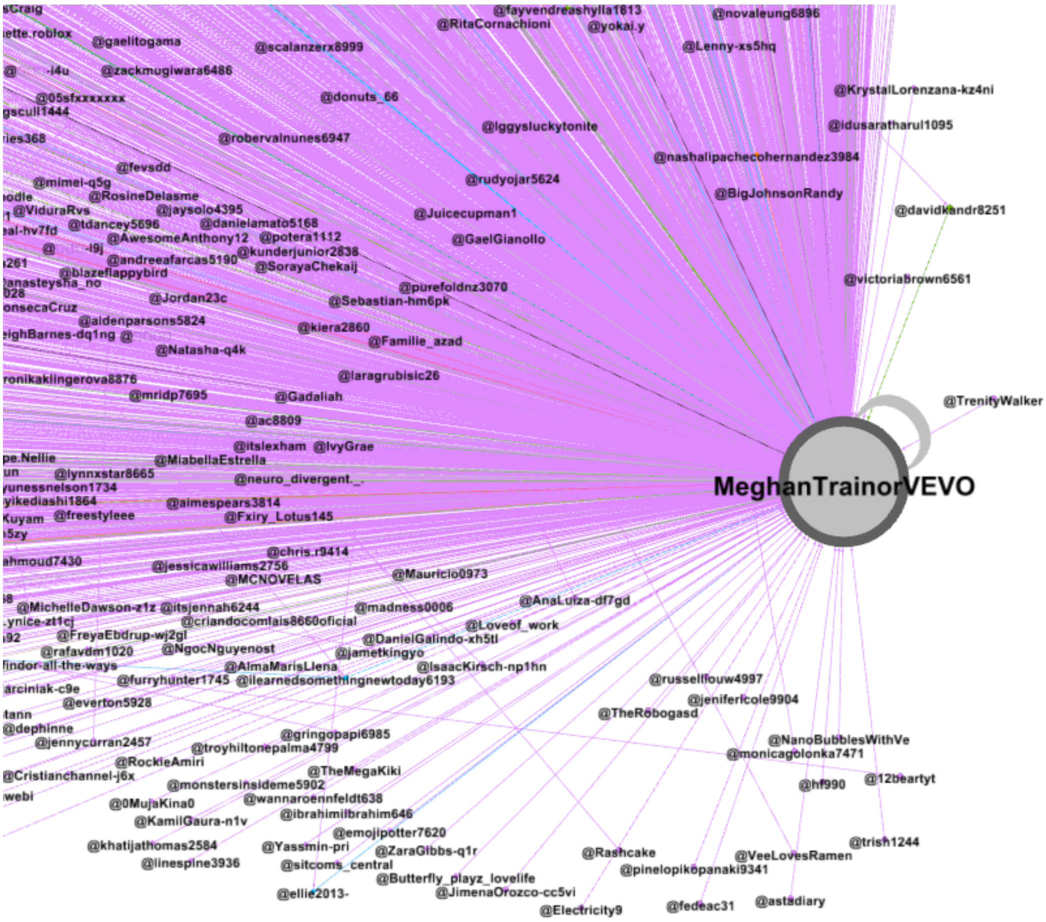
Although @JarataZuibatuDavies has more replies, @crazycatpetera1404 has a higher PageRank because the algorithm focuses on who interacts with you, not just the number of replies. @crazycatpetera1404 likely received replies from more influential users, making them rank higher in the network. This shows that influence in networks depends on the quality of interactions rather than the quantity.

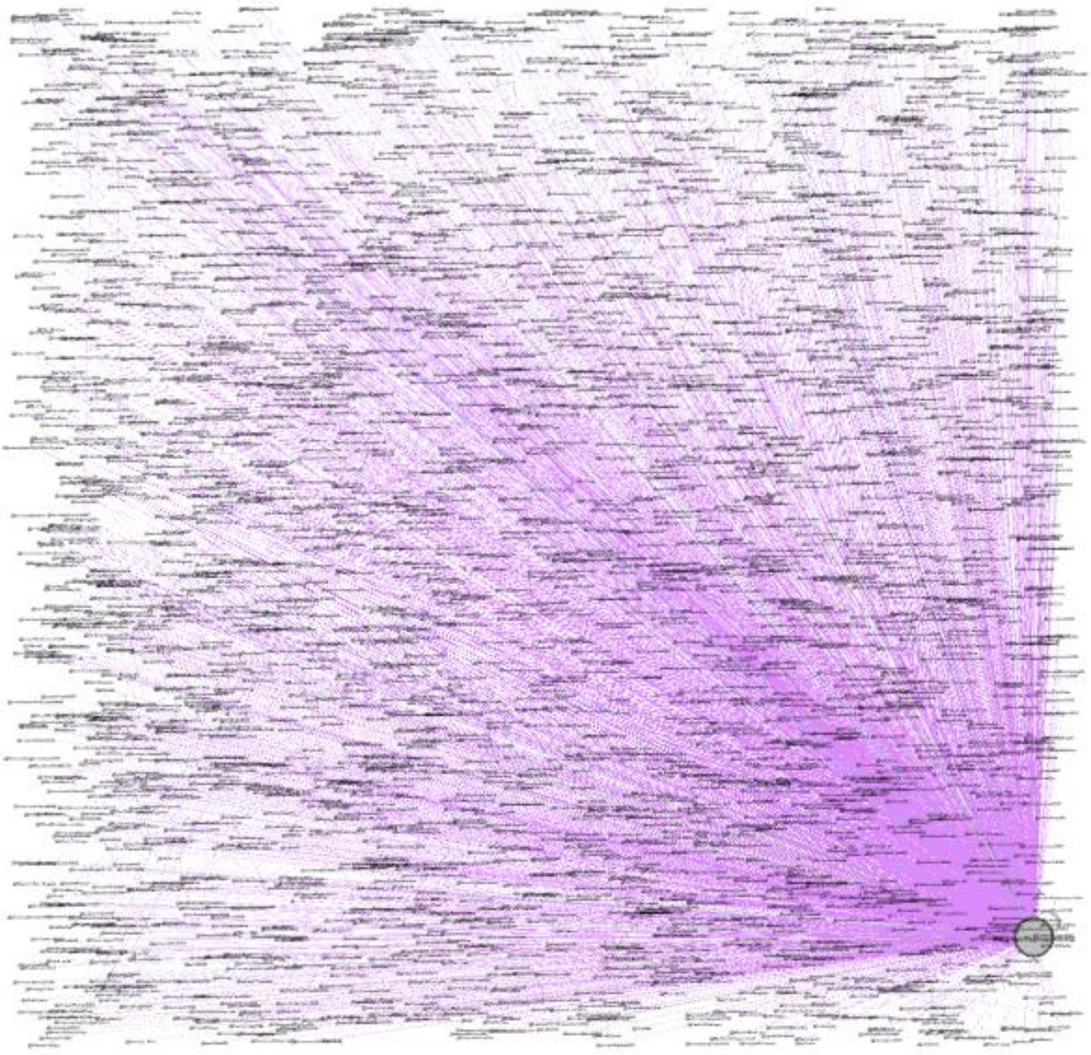
Alternatively, I run the pagerank algorithm on Gephi as well. It seems like the result is slightly different from R-studio that @crazycatpetera is not top1 but change to 7th place, and the reason is that the PageRank algorithm is a non-deterministic algorithm. I also run the PageRank Statistics to create the distribution chart.



screen_name	node_type	PageRank
MeghanTrainorVEVO	publisher	0.120291
	actor	0.001361
@JarataZuibatuDavies	actor	0.000802
@Aidan-wise2009	actor	0.000781
@strawberrycowgirl5987	actor	0.00076
@Smiling3DModel-md6vq	actor	0.000656
@chicken	actor	0.000453
@Alanna_isSLAY	actor	0.000419
@crazycatpetera1404	actor	0.000405
@juanpabloriverazenteno27...	actor	0.000335




sCraig
@fayvendreashylla1613
@novaleung6896
@RitaCornachioni
@yokai.y
iette.roblox
@gaelitogama
@scalanzerx8999
@Lenny-xs5hq
@-i4u
@zackmugiwara6486
@KrystalLorenzana-kz4ni
@05sfxxxxxxx
@donuts_66
gscull1444
@iggysluckytonite
@idusaratharul1095
ries368
@robervalnunes6947
@nashalipachecohernandez3984
@fevsdd
@rudyojar5624
@mimei-q5g
@BigJohnsonRandy
oodle
@RosineDelasme
@ViduraRvs
@jaysolo4395
@Juicecupman1
@davidkandr8251
eal-hv7fd
@tdancey5696
@danielamato5168
@AwesomeAnthony12
@potera1112
@GaelGianollo
@-i9j
@kunderjunior2838
261
@andreeafarcas5190
@SorayaChekaij
@blazeflappybird
@victoriabrown6561
anasteysha_no
028
@purefoldnz3070
@Jordan23c
@Sebastian-hm6pk
onsecaCruz
@aidenparsons5824
@kiera2860
eighBarnes-dq1ng
@
@Familie_azad
@Natasha-q4k
ronikaklingerova8876
@laragrubisic26
@mridp7695
@Gadaliah
@TrenityWalker
@ac8809
@itslexham
@IvyGrae
pe.Nellie
@MiabellaEstrella
un
@lynnxstar8665
@neuro_divergent._.
MeghanTrainorVEVO
yunessnelson1734
yikediashi1864
@aimespears3814
Kuyam
@freestyleee
@Fxiry_Lotus145
5zy
@chris.r9414
ahmoud7430
@jessicawilliams2756
@Mauricio0973
@MCNOVELAS
@AnaLuiza-df7gd
@MichelleDawson-z1z
@itsjennah6244
@madness0006
ynice-zt1cj
@criandocomlais8660oficial
@Loveof_work
92
@FreyaEbdrup-wj2gl
@DanielGalindo-xh5tl
@rafavdm1020
@NgocNguyenost
@jametkingyo
findor-all-the-ways
@AlmaMarisLlena
@IsaacKirsch-np1hn
@furryhunter1745
@ilearnedsomethingnewtoday6193
@russellfouw4997
arciniak-c9e
@everton5928
@jenifericole9904
@TheRobogasd
tann
@dephinne
@gringopapi6985
@NanoBubblesWithVe
@jennycurran2457
@troyhiltonepalma4799
@monicagolonka7471
@RockieAmiri
@TheMegaKiki
@Cristianchannel-j6x
@12beartyt
@monstersinsideme5902
@hf990
webi
@wannaroennfeldt638
@0MujaKina0
@ibrahimilbrahim646
@KamilGaura-n1v
@emojipotter7620
@trish1244
@khatijathomas2584
@Yassmin-pri
@ZaraGibbs-q1r
@Rashcake
@VeeLovesRamen
@linespine3936
@sitcoms_central
@pinelopikopanaki9341
@Butterfly_playz_lovelife
@JimenaOrozco-cc5vi
@ellie2013-
@astadiary
@fedeac31
@Electricity9

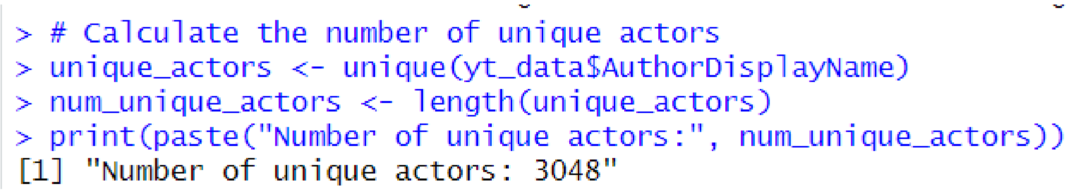


From the graph, there is no obvious big node other than the Maghen Trainer Vevo.

2.3 Unique actors

Calculate how many unique actors there are in your datasets. Explain the code you have used for the calculation. What do the results tell you?

Number of unique actors: 3048



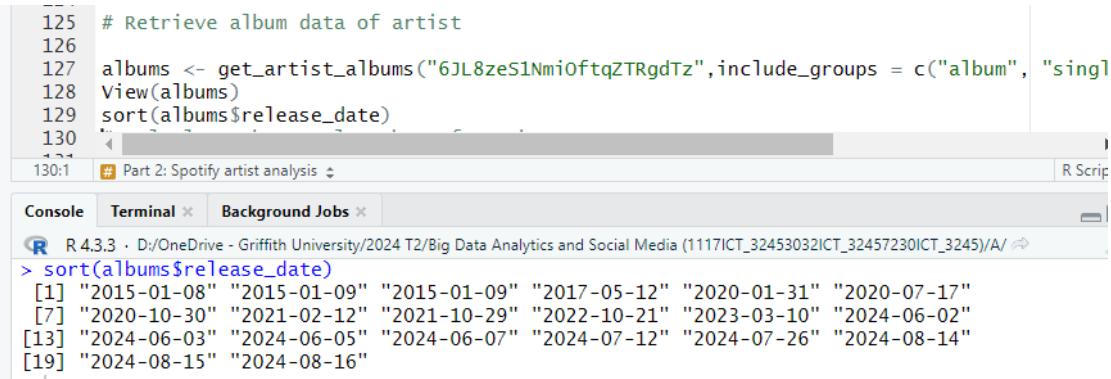
```
> # Calculate the number of unique actors
> unique_actors <- unique(yt_data$AuthorDisplayName)
> num_unique_actors <- length(unique_actors)
> print(paste("Number of unique actors:", num_unique_actors))
[1] "Number of unique actors: 3048"
```

I set the maxComments to 3000, but the number of unique actors was 3048. This is because the maxComments setting limits the number of comments, not unique users. Replies to comments add additional interactions, leading to more unique actors. Some users also post multiple comments but are counted only once.

2.4 Use the Spotify API to extract data

How many years have they been active?

From 2015-01-08 to 2024-08-16. 8 years.

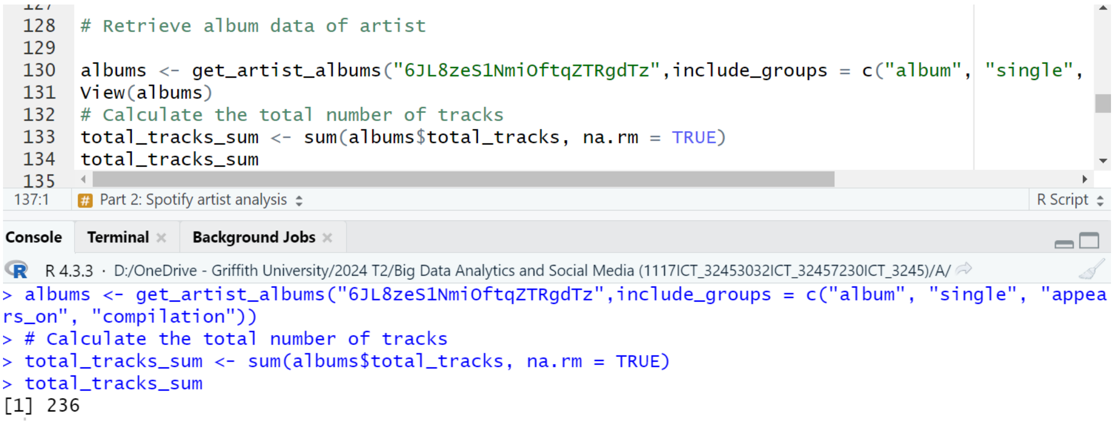

130:1
Part 2: Spotify artist analysis
R Scrip
Console
Terminal
Background Jobs
R 4.3.3 · D:/OneDrive - Griffith University/2024 T2/Big Data Analytics and Social Media (1117ICT_32453032ICT_32457230ICT_3245)/A/

```
# Retrieve album data of artist

albums <- get_artist_albums("6JL8zeS1NmiOftqZTRgdTz",include_groups = c("album", "singl
View(albums)
sort(albums$release_date)
```

```
> sort(albums$release_date)
 [1] "2015-01-08" "2015-01-09" "2015-01-09" "2017-05-12" "2020-01-31" "2020-07-17"
 [7] "2020-10-30" "2021-02-12" "2021-10-29" "2022-10-21" "2023-03-10" "2024-06-02"
[13] "2024-06-03" "2024-06-05" "2024-06-07" "2024-07-12" "2024-07-26" "2024-08-14"
[19] "2024-08-15" "2024-08-16"
```

○ How many albums & songs have they published?

20 albums, 236 songs(tracks).

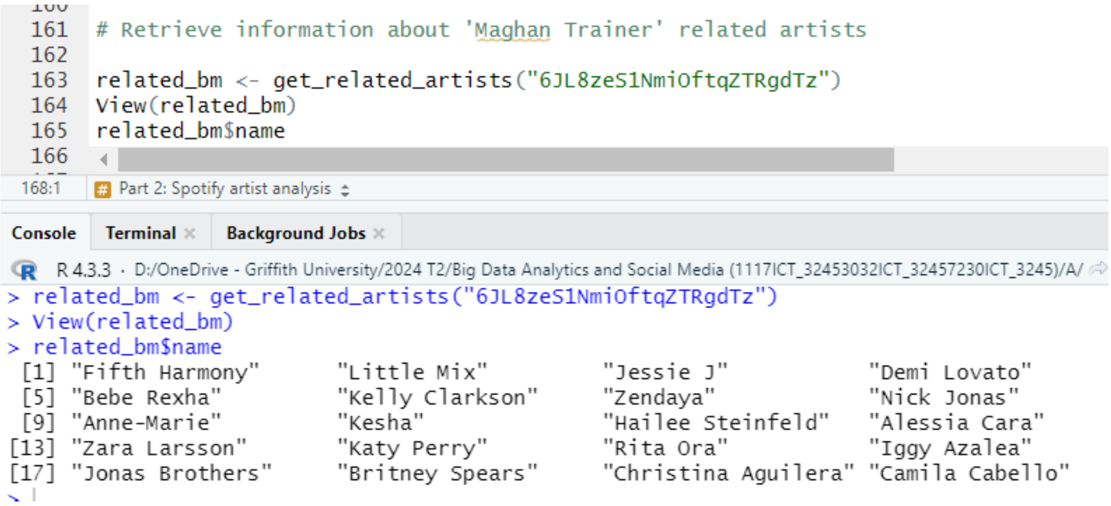

137:1
Part 2: Spotify artist analysis
R Script
Console
Terminal
Background Jobs
R 4.3.3 · D:/OneDrive - Griffith University/2024 T2/Big Data Analytics and Social Media (1117ICT_32453032ICT_32457230ICT_3245)/A/

```
# Retrieve album data of artist

albums <- get_artist_albums("6JL8zeS1NmiOftqZTRgdTz",include_groups = c("album", "single",
View(albums)
# Calculate the total number of tracks
total_tracks_sum <- sum(albums$total_tracks, na.rm = TRUE)
total_tracks_sum
```

```
> albums <- get_artist_albums("6JL8zeS1NmiOftqZTRgdTz",include_groups = c("album", "single", "appea
rs_on", "compilation"))
> # Calculate the total number of tracks
> total_tracks_sum <- sum(albums$total_tracks, na.rm = TRUE)
> total_tracks_sum
[1] 236
```

○ With whom have they often collaborated?

Fifth Harmony, Little Mix, Jessie J.... Camila Cabello.

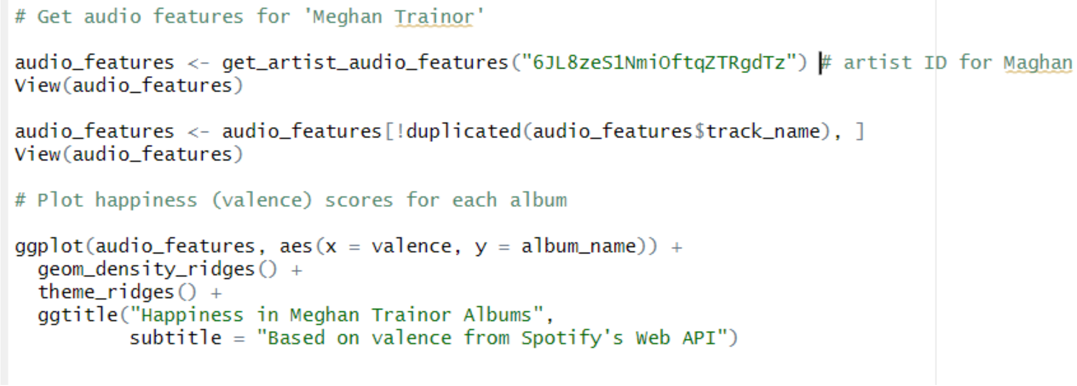


```
# Retrieve information about 'Maghan Trainer' related artists

related_bm <- get_related_artists("6JL8zeS1NmiOftqZTRgdTz")
View(related_bm)
related_bm$name
```

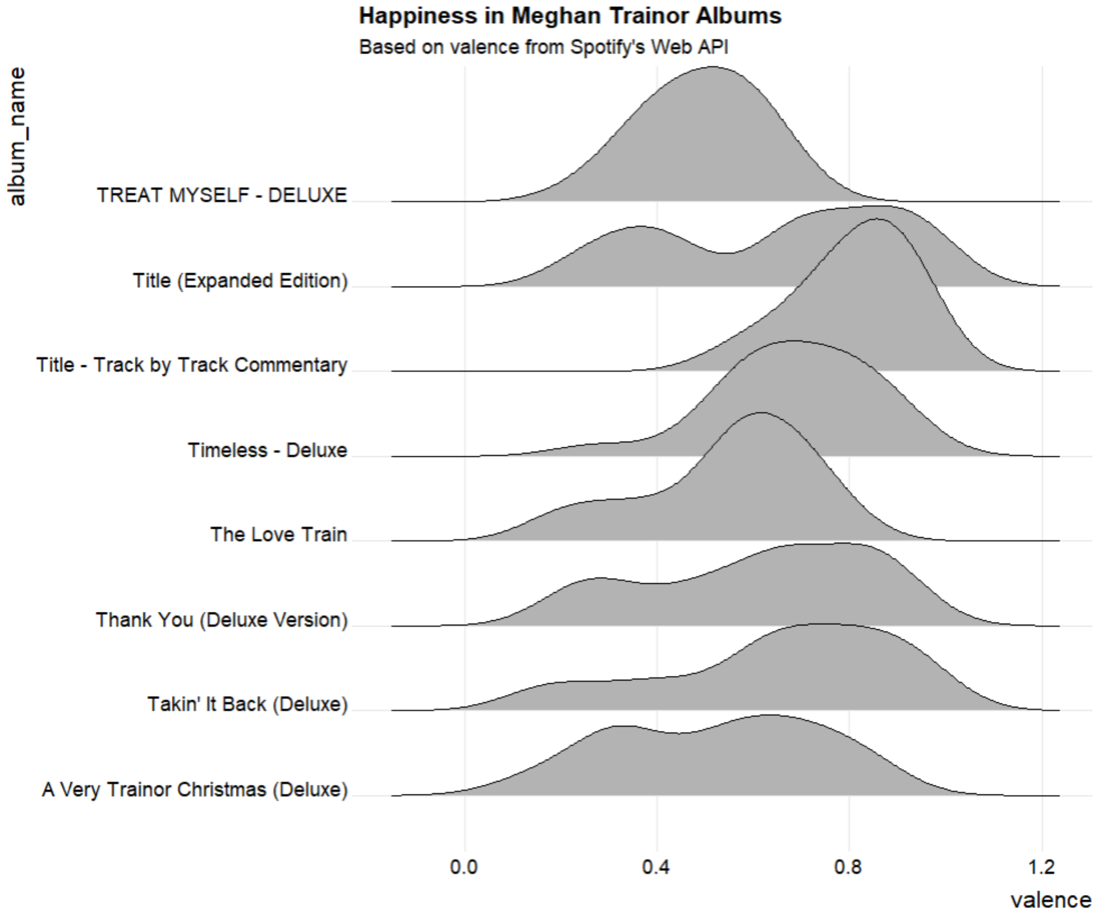
```
R 4.3.3 · D:/OneDrive - Griffith University/2024 T2/Big Data Analytics and Social Media (1117ICT_32453032ICT_32457230ICT_3245)/A/
> related_bm <- get_related_artists("6JL8zeS1NmiOftqZTRgdTz")
> View(related_bm)
> related_bm$name
 [1] "Fifth Harmony"     "Little Mix"        "Jessie J"          "Demi Lovato"
 [5] "Bebe Rexha"        "Kelly Clarkson"    "Zendaya"           "Nick Jonas"
 [9] "Anne-Marie"        "Kesha"             "Hailee Steinfeld"  "Alessia Cara"
[13] "Zara Larsson"      "Katy Perry"        "Rita Ora"          "Iggy Azalea"
[17] "Jonas Brothers"    "Britney Spears"    "Christina Aguilera" "Camila Cabello"
> 
```

2.4 Revalent features/ valence of Meghan Trainor’s songs

Revalent features of their songs (e.g., valence)?



```
# Get audio features for 'Meghan Trainor'

audio_features <- get_artist_audio_features("6JL8zeS1NmiOftqZTRgdTz") # artist ID for Maghan
View(audio_features)

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

# Plot happiness (valence) scores for each album

ggplot(audio_features, aes(x = valence, y = album_name)) +
  geom_density_ridges() +
  theme_ridges() +
  ggtitle("Happiness in Meghan Trainor Albums",
          subtitle = "Based on valence from Spotify's Web API")
```



How does the Spotify data compare to the information you collected from other sources in Question 1)?

The Spotify data shows that Meghan Trainor has been active since 2015, with 20 albums and 236 tracks. This differs from other sources like Wikipedia, which states she started in 2014 and has released five studio albums (Wikipedia, 2024). The discrepancy may be due to Spotify including compilations or live albums. Both sources agree on her major collaborations, but Spotify adds insights into her musical style through audio features.

3. Text Pre-Processing

3.1 Reddit Posts Term-Document Matrices

After performing text pre-processing, create Term-Document Matrices for your data. What are the 10 terms occurring with the highest frequency? Explain the results.

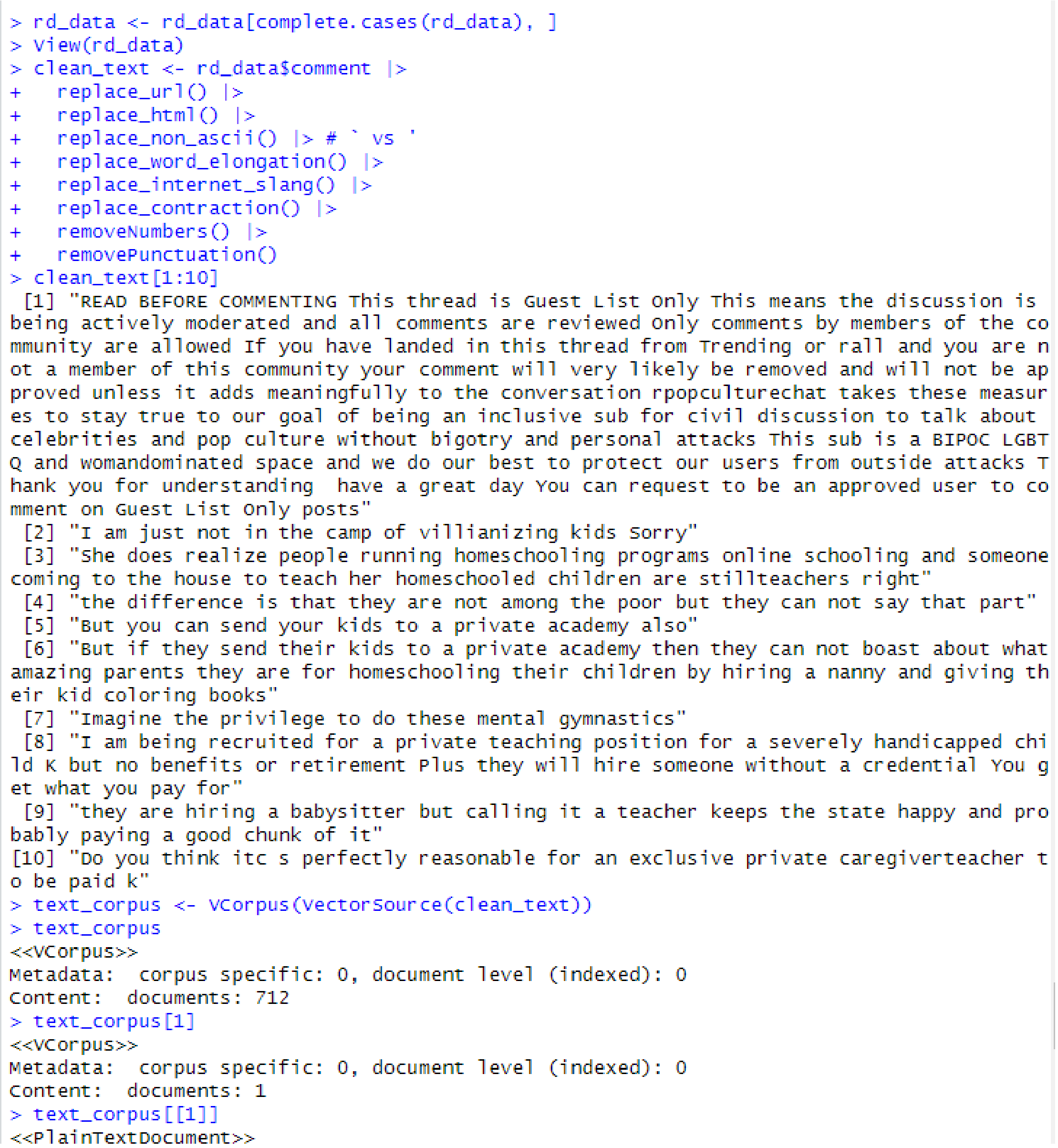
3.2 Top10 highest frequency terms

I used 2 Reddit thread about Meghan Trainer to perfrom Term-Document Matrices. The 10 most frequent terms are: "teachers," "school," "people," "fuck," "Meghan," "teacher," "kids," "public," "music," and "Trisha." These terms suggest discussions centered around education, strong emotional reactions, and Meghan Trainor's public comments, highlighting both her personal actions and her music career.

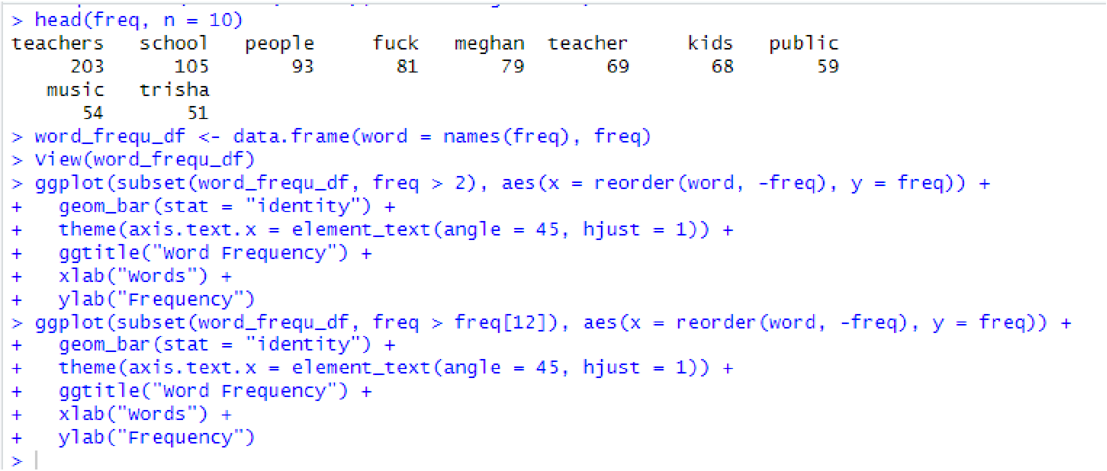
Thread 1:
https://www.reddit.com/r/popculturechat/comments/12womcl/meghan_trainor_says_fck_teachers_and_slams_public/

Thread2:

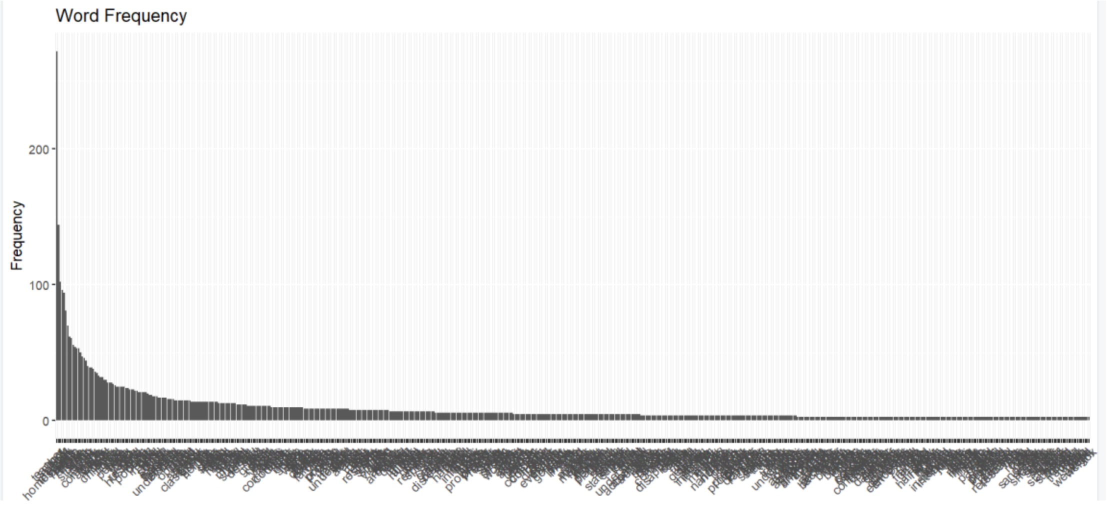
https://www.reddit.com/r/Teachers/comments/12wft54/meghan_trainor_has_lost_all_of_my_respect/

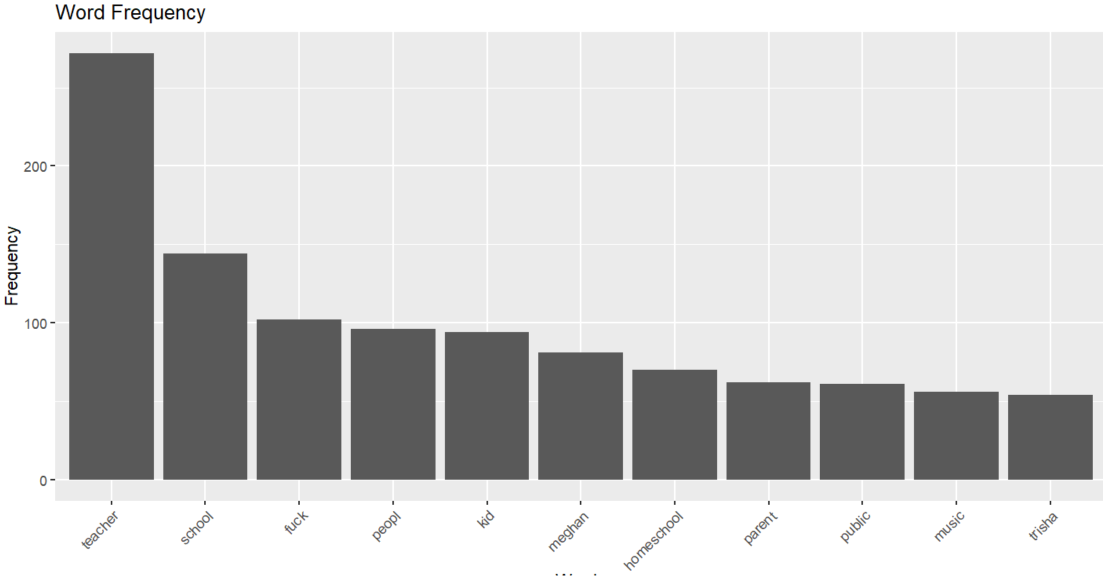


```
> rd_data <- rd_data[complete.cases(rd_data), ]
> View(rd_data)
> clean_text <- rd_data$comment |>
+   replace_url() |>
+   replace_html() |>
+   replace_non_ascii() |> # ` vs '
+   replace_word_elongation() |>
+   replace_internet_slang() |>
+   replace_contraction() |>
+   removeNumbers() |>
+   removePunctuation()
> clean_text[1:10]
 [1] "READ BEFORE COMMENTING This thread is Guest List Only This means the discussion is 
being actively moderated and all comments are reviewed Only comments by members of the co
mmunity are allowed If you have landed in this thread from Trending or rall and you are n
ot a member of this community your comment will very likely be removed and will not be ap
proved unless it adds meaningfully to the conversation rpopculturechat takes these measur
es to stay true to our goal of being an inclusive sub for civil discussion to talk about 
celebrities and pop culture without bigotry and personal attacks This sub is a BIPOC LGBT
Q and womandominated space and we do our best to protect our users from outside attacks T
hank you for understanding  have a great day You can request to be an approved user to co
mment on Guest List Only posts"
 [2] "I am just not in the camp of villianizing kids Sorry"
 [3] "She does realize people running homeschooling programs online schooling and someone
coming to the house to teach her homeschooled children are stillteachers right"
 [4] "the difference is that they are not among the poor but they can not say that part"
 [5] "But you can send your kids to a private academy also"
 [6] "But if they send their kids to a private academy then they can not boast about what
amazing parents they are for homeschooling their children by hiring a nanny and giving th
eir kid coloring books"
 [7] "Imagine the privilege to do these mental gymnastics"
 [8] "I am being recruited for a private teaching position for a severely handicapped chi
ld K but no benefits or retirement Plus they will hire someone without a credential You g
et what you pay for"
 [9] "they are hiring a babysitter but calling it a teacher keeps the state happy and pro
bably paying a good chunk of it"
[10] "Do you think itc s perfectly reasonable for an exclusive private caregiverteacher t
o be paid k"
> text_corpus <- VCorpus(VectorSource(clean_text))
> text_corpus
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 712
> text_corpus[1]
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 1
> text_corpus[[1]]
<<PlainTextDocument>>
```



```
> head(freq, n = 10)
teachers   school   people     fuck   meghan  teacher     kids   public 
     203      105       93       81       79       69       68       59 
   music   trisha 
      54       51 
> word_frequ_df <- data.frame(word = names(freq), freq)
> View(word_frequ_df)
> ggplot(subset(word_frequ_df, freq > 2), aes(x = reorder(word, -freq), y = freq)) +
+   geom_bar(stat = "identity") +
+   theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
+   ggtitle("Word Frequency") +
+   xlab("Words") +
+   ylab("Frequency")
> ggplot(subset(word_frequ_df, freq > freq[12]), aes(x = reorder(word, -freq), y = freq)) +
+   geom_bar(stat = "identity") +
+   theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
+   ggtitle("Word Frequency") +
+   xlab("Words") +
+   ylab("Frequency")
> |
```

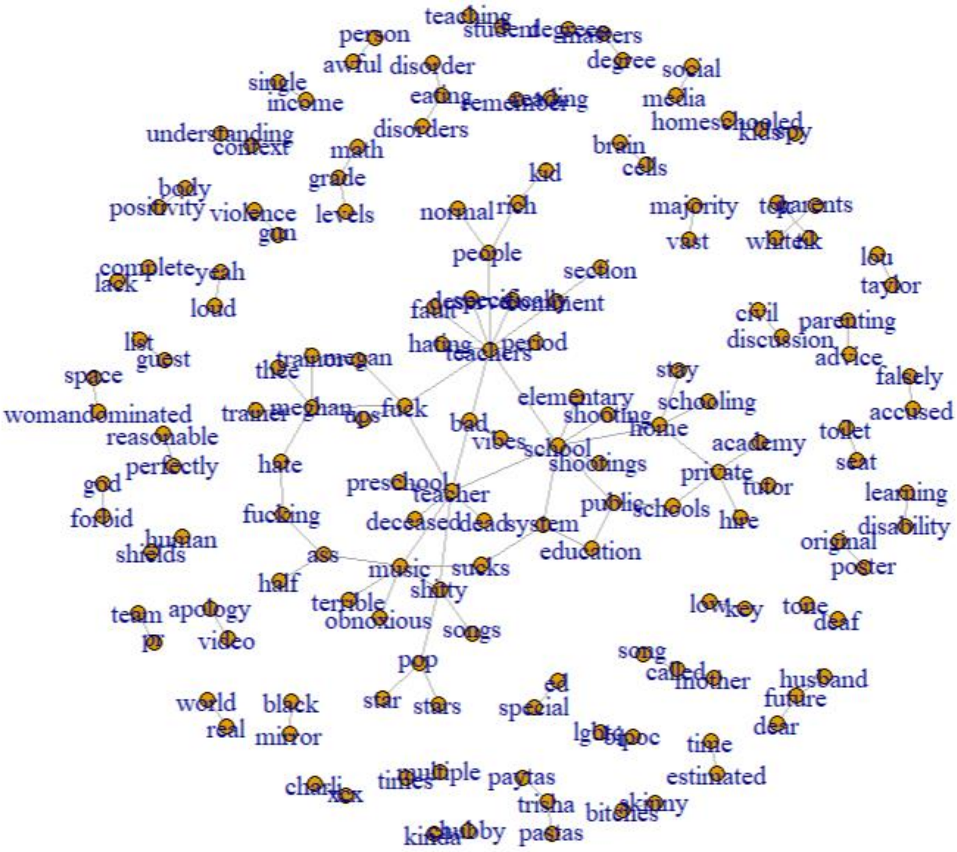


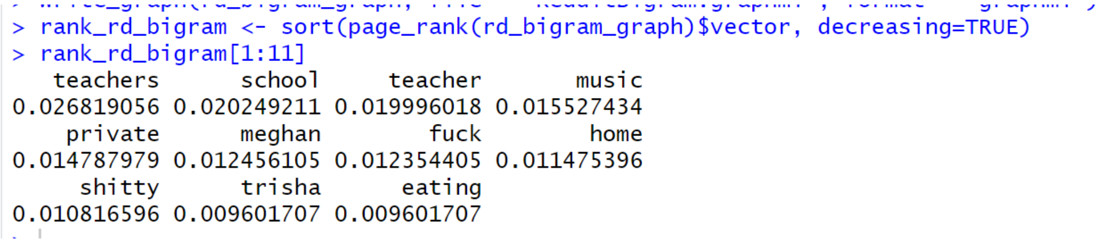


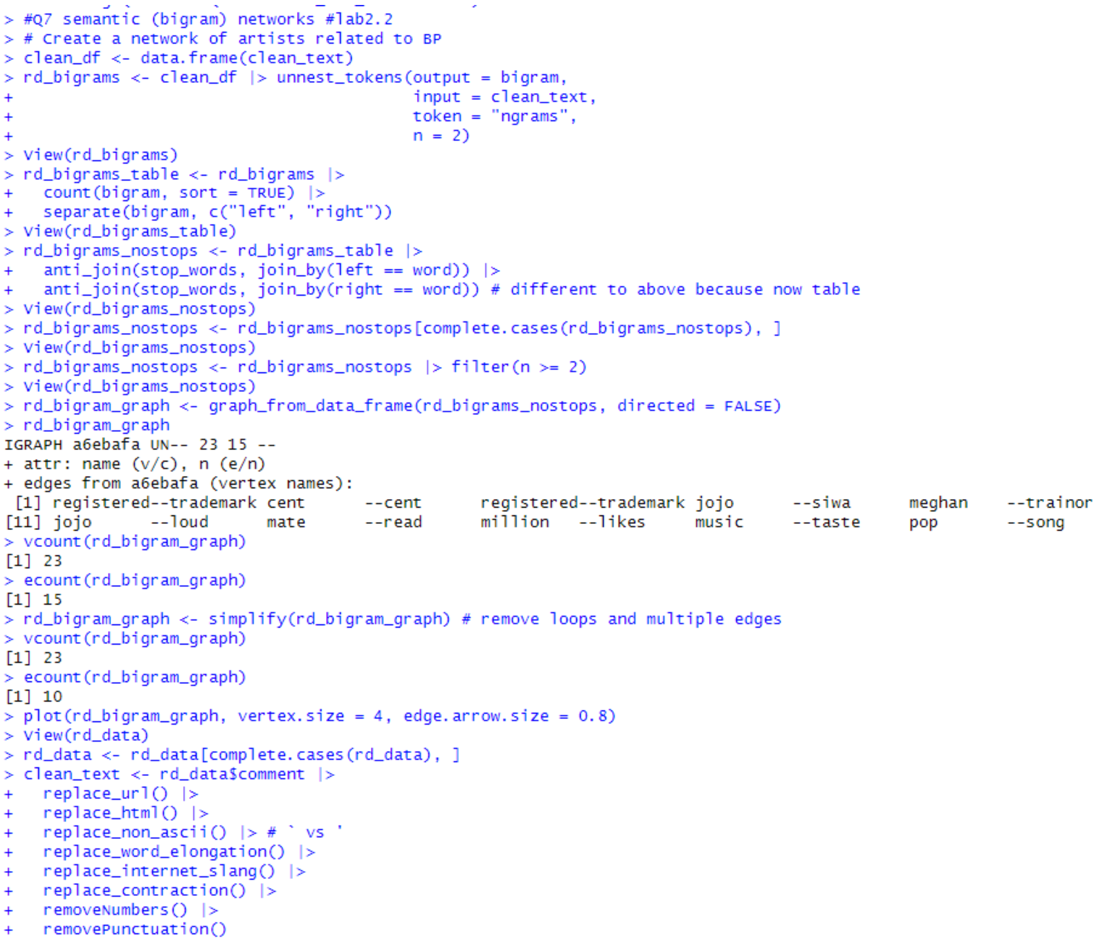
3.2 Page rank semantic (bigram) networks

After performing text pre-processing, create semantic (bigram) networks from your data and list the top 10 most important terms according to page rank. Explain how and why they differ from your results for the question above. [2 marks]

The top 10 most important terms from the bigram network by PageRank are: "teachers," "school," "teacher," "music," "private," "Meghan," "fuck," "home," "shitty," and "Trisha." These terms differ from the unigrams in the previous analysis, as the bigram network captures relationships between words, revealing connections in the discussions. For example, "teachers" and "school" remain prominent due to educational topics, but now "private" and "home" emerge, reflecting deeper discourse around homeschooling. The presence of more emotional or vulgar terms like "fuck" and "shitty" shows stronger expressions of sentiment when terms are paired together in context, explaining their higher influence in the semantic network.




```
> write_graph(rd_bigram_graph, file = "RedditBigram.graphml", format = "graphml")
> rank_rd_bigram <- sort(page_rank(rd_bigram_graph)$vector, decreasing=TRUE)
> rank_rd_bigram[1:11]
   teachers      school     teacher       music 
0.026819056 0.020249211 0.019996018 0.015527434 
    private      meghan        fuck        home 
0.014787979 0.012456105 0.012354405 0.011475396 
     shitty      trisha      eating 
0.010816596 0.009601707 0.009601707 
>
```



```
> #Q7 semantic (bigram) networks #lab2.2
> # Create a network of artists related to BP
> clean_df <- data.frame(clean_text)
> rd_bigrams <- clean_df |> unnest_tokens(output = bigram,
+                                         input = clean_text,
+                                         token = "ngrams",
+                                         n = 2)
> View(rd_bigrams)
> rd_bigrams_table <- rd_bigrams |>
+   count(bigram, sort = TRUE) |>
+   separate(bigram, c("left", "right"))
> View(rd_bigrams_table)
> rd_bigrams_nostops <- rd_bigrams_table |>
+   anti_join(stop_words, join_by(left == word)) |>
+   anti_join(stop_words, join_by(right == word)) # different to above because now table
> View(rd_bigrams_nostops)
> rd_bigrams_nostops <- rd_bigrams_nostops[complete.cases(rd_bigrams_nostops), ]
> View(rd_bigrams_nostops)
> rd_bigrams_nostops <- rd_bigrams_nostops |> filter(n >= 2)
> View(rd_bigrams_nostops)
> rd_bigram_graph <- graph_from_data_frame(rd_bigrams_nostops, directed = FALSE)
> rd_bigram_graph
IGRAPH a6ebafa UN-- 23 15 --
+ attr: name (v/c), n (e/n)
+ edges from a6ebafa (vertex names):
 [1] registered--trademark cent      --cent      registered--trademark jojo      --siwa      meghan    --trainor
[11] jojo      --loud      mate      --read      million   --likes     music     --taste     pop       --song
> vcount(rd_bigram_graph)
[1] 23
> ecount(rd_bigram_graph)
[1] 15
> rd_bigram_graph <- simplify(rd_bigram_graph) # remove loops and multiple edges
> vcount(rd_bigram_graph)
[1] 23
> ecount(rd_bigram_graph)
[1] 10
> plot(rd_bigram_graph, vertex.size = 4, edge.arrow.size = 0.8)
> View(rd_data)
> rd_data <- rd_data[complete.cases(rd_data), ]
> clean_text <- rd_data$comment |>
+   replace_url() |>
+   replace_html() |>
+   replace_non_ascii() |> # ` vs '
+   replace_word_elongation() |>
+   replace_internet_slang() |>
+   replace_contraction() |>
+   removeNumbers() |>
+   removePunctuation()
```

4. Social Network Analysis

4.1 Centrality analysis: Degree, Betweenness, and Closeness

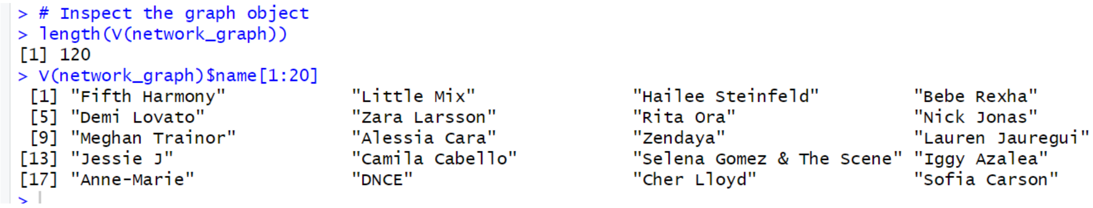
Perform centrality analysis by detecting degree centrality, betweenness centrality, and closeness centrality. Explain how relevant the results are to your artist/band.

The network contains 120 nodes (artists)

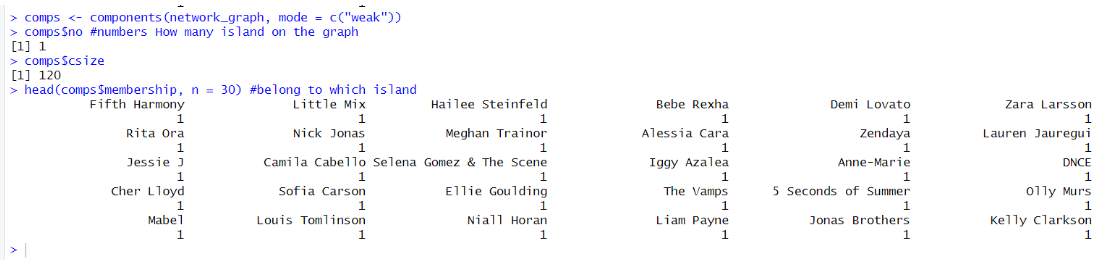
Example artist: Fifth Harmony, Little Mix, Meghan Trainor, Bebe Rexha...

The entire graph is weakly connected (all nodes are in one connected component). Since the graph is fully connected, this component includes all 120 artists.

Centrality Analysis:



```
> # Inspect the graph object
> length(V(network_graph))
[1] 120
> V(network_graph)$name[1:20]
 [1] "Fifth Harmony"            "Little Mix"               "Hailee Steinfeld"         "Bebe Rexha"
 [5] "Demi Lovato"              "Zara Larsson"             "Rita Ora"                 "Nick Jonas"
 [9] "Meghan Trainor"           "Alessia Cara"             "Zendaya"                  "Lauren Jauregui"
[13] "Jessie J"                 "Camila Cabello"           "Selena Gomez & The Scene" "Iggy Azalea"
[17] "Anne-Marie"               "DNCE"                     "Cher Lloyd"               "Sofia Carson"
> 
```



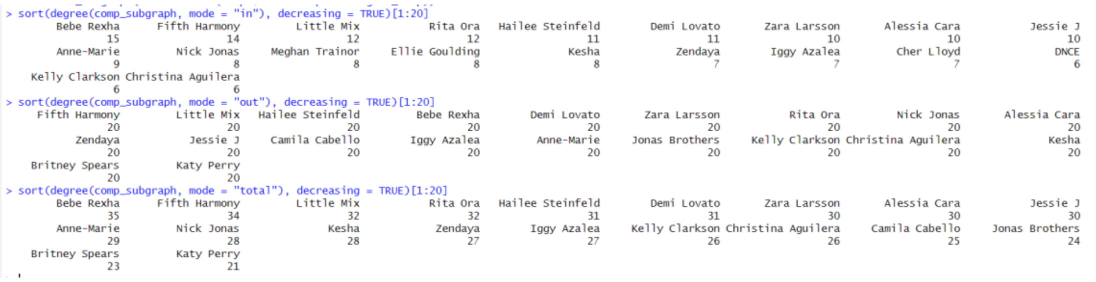
```
> comps <- components(network_graph, mode = c("weak"))
> comps$no #numbers How many island on the graph
[1] 1
> comps$csize
[1] 120
> head(comps$membership, n = 30) #belong to which island
           Fifth Harmony               Little Mix         Hailee Steinfeld               Bebe Rexha              Demi Lovato             Zara Larsson
                       1                        1                        1                        1                        1                        1
                Rita Ora               Nick Jonas           Meghan Trainor             Alessia Cara                  Zendaya          Lauren Jauregui
                       1                        1                        1                        1                        1                        1
                Jessie J           Camila Cabello Selena Gomez & The Scene              Iggy Azalea               Anne-Marie                     DNCE
                       1                        1                        1                        1                        1                        1
              Cher Lloyd             Sofia Carson           Ellie Goulding                The Vamps      5 Seconds of Summer                Olly Murs
                       1                        1                        1                        1                        1                        1
                   Mabel          Louis Tomlinson              Niall Horan               Liam Payne           Jonas Brothers           Kelly Clarkson
                       1                        1                        1                        1                        1                        1
> 
```

Degree Centrality:

In-degree: Bebe Rexha has the highest in-degree with 15 connections, meaning that 15 other artists are related to her, and Fifth Harmony(14), Little Mix(12), Rita Ora(12) also have high in-degrees.

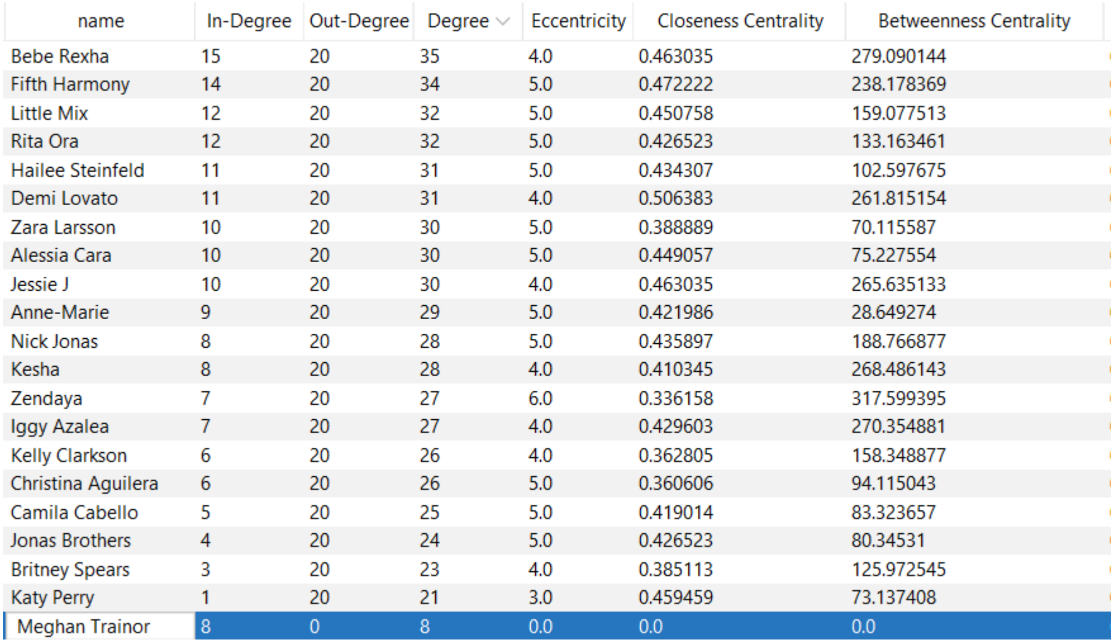
Out-degree: Fifth Harmony, Little Mix, Hailee Steinfeld, and Bebe Rexha, have an out-degree of 20, indicating that they are related to many other artists.

Total degree: Bebe Rexha has the highest total degree with 35, making her the most connected artist

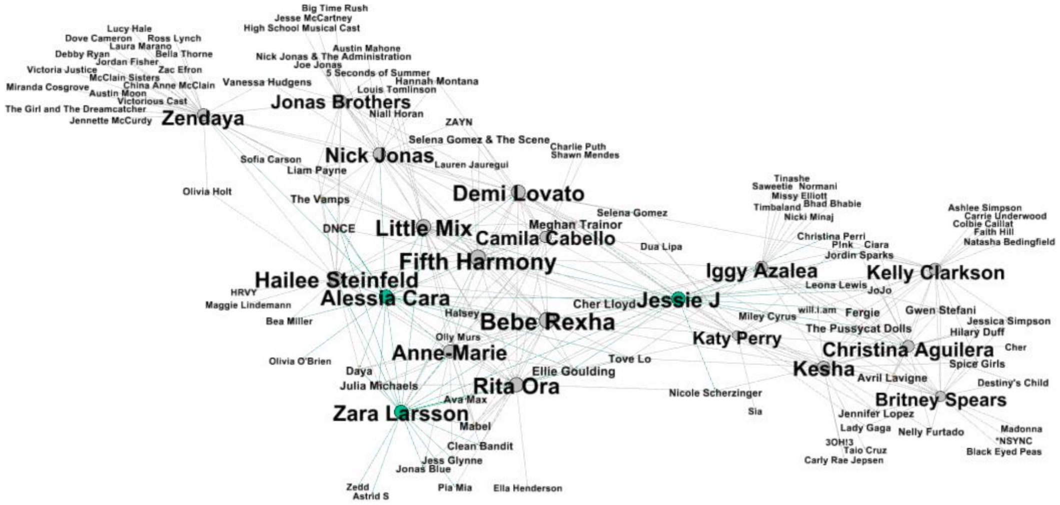


```
> sort(degree(comp_subgraph, mode = "in"), decreasing = TRUE)[1:20]
        Bebe Rexha      Fifth Harmony         Little Mix           Rita Ora   Hailee Steinfeld        Demi Lovato       Zara Larsson       Alessia Cara           Jessie J
                15                 14                 12                 12                 11                 11                 10                 10                 10
        Anne-Marie         Nick Jonas     Meghan Trainor     Ellie Goulding              Kesha            Zendaya        Iggy Azalea         Cher Lloyd               DNCE
                 9                  8                  8                  8                  8                  7                  7                  7                  6
    Kelly Clarkson Christina Aguilera
                 6                  6
> sort(degree(comp_subgraph, mode = "out"), decreasing = TRUE)[1:20]
     Fifth Harmony         Little Mix   Hailee Steinfeld         Bebe Rexha        Demi Lovato       Zara Larsson           Rita Ora         Nick Jonas       Alessia Cara
                20                 20                 20                 20                 20                 20                 20                 20                 20
           Zendaya           Jessie J     Camila Cabello        Iggy Azalea         Anne-Marie     Jonas Brothers     Kelly Clarkson Christina Aguilera              Kesha
                20                 20                 20                 20                 20                 20                 20                 20                 20
    Britney Spears         Katy Perry
                20                 20
> sort(degree(comp_subgraph, mode = "total"), decreasing = TRUE)[1:20]
        Bebe Rexha      Fifth Harmony         Little Mix           Rita Ora   Hailee Steinfeld        Demi Lovato       Zara Larsson       Alessia Cara           Jessie J
                35                 34                 32                 32                 31                 31                 30                 30                 30
        Anne-Marie         Nick Jonas              Kesha            Zendaya        Iggy Azalea     Kelly Clarkson Christina Aguilera     Camila Cabello     Jonas Brothers
                29                 28                 28                 27                 27                 26                 26                 25                 24
    Britney Spears         Katy Perry
                23                 21
```

The output report using Gephi

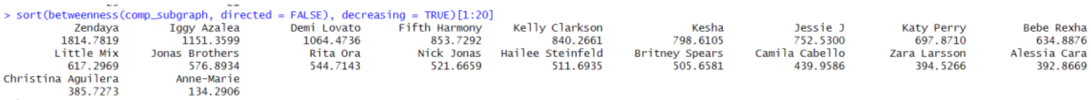


name	In-Degree	Out-Degree	Degree	Eccentricity	Closeness Centrality	Betweenness Centrality
Bebe Rexha	15	20	35	4.0	0.463035	279.090144
Fifth Harmony	14	20	34	5.0	0.472222	238.178369
Little Mix	12	20	32	5.0	0.450758	159.077513
Rita Ora	12	20	32	5.0	0.426523	133.163461
Hailee Steinfeld	11	20	31	5.0	0.434307	102.597675
Demi Lovato	11	20	31	4.0	0.506383	261.815154
Zara Larsson	10	20	30	5.0	0.388889	70.115587
Alessia Cara	10	20	30	5.0	0.449057	75.227554
Jessie J	10	20	30	4.0	0.463035	265.635133
Anne-Marie	9	20	29	5.0	0.421986	28.649274
Nick Jonas	8	20	28	5.0	0.435897	188.766877
Kesha	8	20	28	4.0	0.410345	268.486143
Zendaya	7	20	27	6.0	0.336158	317.599395
Iggy Azalea	7	20	27	4.0	0.429603	270.354881
Kelly Clarkson	6	20	26	4.0	0.362805	158.348877
Christina Aguilera	6	20	26	5.0	0.360606	94.115043
Camila Cabello	5	20	25	5.0	0.419014	83.323657
Jonas Brothers	4	20	24	5.0	0.426523	80.34531
Britney Spears	3	20	23	4.0	0.385113	125.972545
Katy Perry	1	20	21	3.0	0.459459	73.137408
Meghan Trainor	8	0	8	0.0	0.0	0.0



The degree graph.

Betweenness Centrality: This means how often an artist acts as a bridge between other artists in the network. Zendaya has the highest betweenness centrality, meaning she is often a bridge connecting different artists and Iggy Azalea and Demi Lovato also have high betweenness.



```
> sort(betweenness(comp_subgraph, directed = FALSE), decreasing = TRUE)[1:20]
           Zendaya        Iggy Azalea        Demi Lovato      Fifth Harmony     Kelly Clarkson              Kesha            Jessie J         Katy Perry         Bebe Rexha 
         1814.7819          1151.3599          1064.4736           853.7292           840.2661           798.6105           752.5300           697.8710           634.8876 
        Little Mix     Jonas Brothers           Rita Ora         Nick Jonas   Hailee Steinfeld     Britney Spears     Camila Cabello       Zara Larsson       Alessia Cara 
          617.2969           576.8934           544.7143           521.6659           511.6935           505.6581           439.9586           394.5266           392.8669 
Christina Aguilera         Anne-Marie 
          385.7273           134.2906 
```

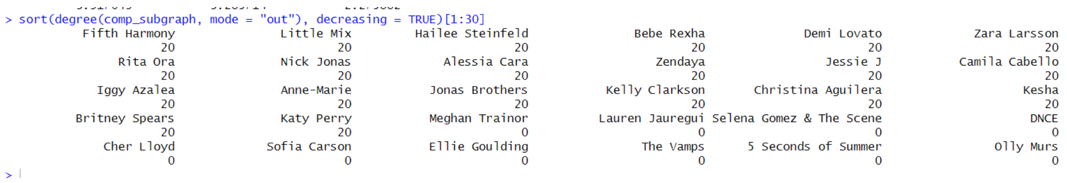
What are the actual degree, betweenness, and closeness centrality scores for your artist/band node in the network?

For Meghan Trainor

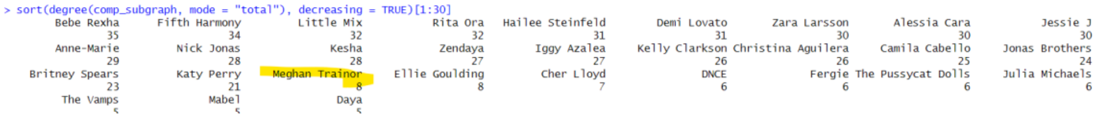
In-degree: 8, out:0, Total:8

Betweenness: 6.619747

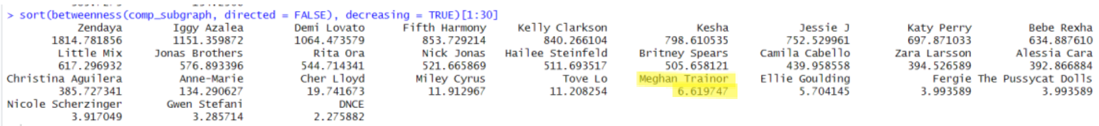
closeness: In: 0.03, out:NA, total:0.034



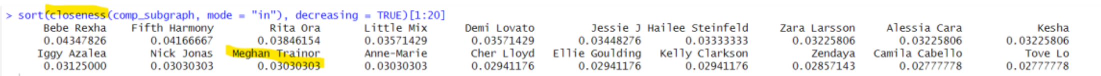
```
> sort(degree(comp_subgraph, mode = "out"), decreasing = TRUE)[1:30]
  Fifth Harmony        Little Mix   Hailee Steinfeld          Bebe Rexha          Demi Lovato        Zara Larsson
             20                20                 20                  20                   20                  20
       Rita Ora        Nick Jonas       Alessia Cara             Zendaya             Jessie J      Camila Cabello
             20                20                 20                  20                   20                  20
   Iggy Azalea         Anne-Marie     Jonas Brothers       Kelly Clarkson    Christina Aguilera               Kesha
             20                20                 20                  20                   20                  20
 Britney Spears        Katy Perry     Meghan Trainor     Lauren Jauregui Selena Gomez & The Scene              DNCE
             20                20                  0                   0                    0                   0
     Cher Lloyd      Sofia Carson      Ellie Goulding           The Vamps    5 Seconds of Summer          Olly Murs
              0                 0                  0                   0                    0                   0
> |
```



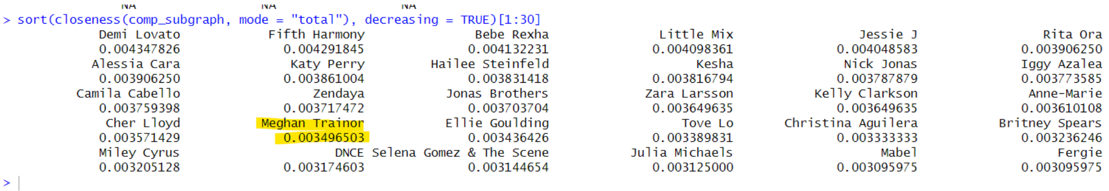
```
> sort(degree(comp_subgraph, mode = "total"), decreasing = TRUE)[1:30]
    Bebe Rexha      Fifth Harmony         Little Mix           Rita Ora   Hailee Steinfeld        Demi Lovato       Zara Larsson       Alessia Cara           Jessie J
            35                 34                 32                 32                 31                 31                 30                 30                 30
    Anne-Marie         Nick Jonas              Kesha            Zendaya        Iggy Azalea     Kelly Clarkson Christina Aguilera     Camila Cabello     Jonas Brothers
            29                 28                 28                 27                 27                 26                 26                 25                 24
Britney Spears         Katy Perry     Meghan Trainor     Ellie Goulding         Cher Lloyd               DNCE Fergie The Pussycat Dolls      Julia Michaels
            23                 21                  8                  8                  7                  6                  6                  6                  6
     The Vamps              Mabel               Daya
             5                  5                  5
```



```
> sort(betweenness(comp_subgraph, directed = FALSE), decreasing = TRUE)[1:30]
           Zendaya        Iggy Azalea        Demi Lovato      Fifth Harmony     Kelly Clarkson              Kesha           Jessie J         Katy Perry         Bebe Rexha
       1814.781856        1151.359872        1064.473579         853.729214         840.266104         798.610535         752.529961         697.871033         634.887610
        Little Mix     Jonas Brothers           Rita Ora         Nick Jonas   Hailee Steinfeld     Britney Spears     Camila Cabello       Zara Larsson       Alessia Cara
        617.296932         576.893396         544.714341         521.665869         511.693517         505.658121         439.958558         394.526589         392.866884
Christina Aguilera         Anne-Marie         Cher Lloyd        Miley Cyrus            Tove Lo     Meghan Trainor     Ellie Goulding             Fergie The Pussycat Dolls
        385.727341         134.290627          19.741673          11.912967          11.208254           6.619747           5.704145           3.993589           3.993589
Nicole Scherzinger       Gwen Stefani               DNCE
          3.917049           3.285714           2.275882
```



```
> sort(closeness(comp_subgraph, mode = "in"), decreasing = TRUE)[1:20]
    Bebe Rexha  Fifth Harmony       Rita Ora     Little Mix    Demi Lovato       Jessie J Hailee Steinfeld  Zara Larsson   Alessia Cara          Kesha
    0.04347826     0.04166667     0.03846154     0.03571429     0.03571429     0.03448276     0.03333333     0.03225806     0.03225806     0.03225806
   Iggy Azalea     Nick Jonas Meghan Trainor     Anne-Marie     Cher Lloyd Ellie Goulding Kelly Clarkson       Zendaya Camila Cabello        Tove Lo
    0.03125000     0.03030303     0.03030303     0.03030303     0.02941176     0.02941176     0.02941176     0.02857143     0.02777778     0.02777778
```



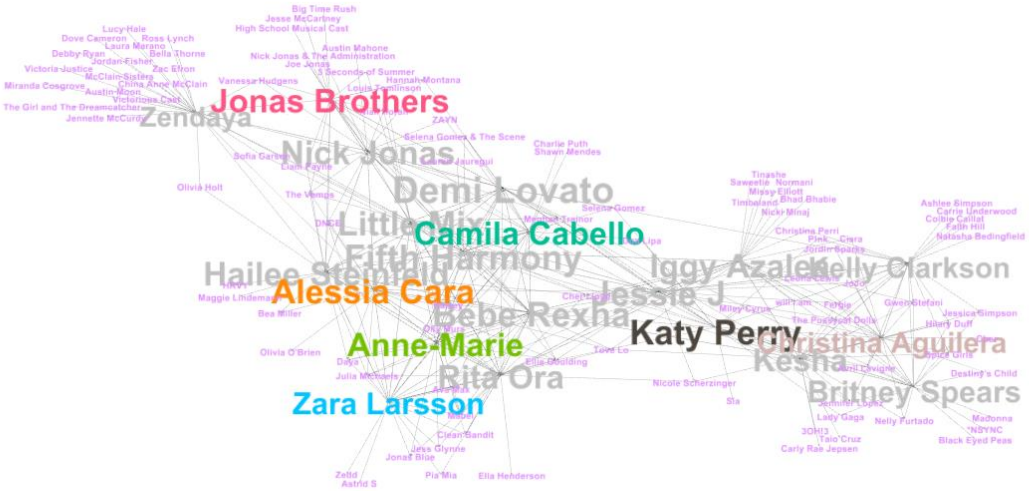
```
> sort(closeness(comp_subgraph, mode = "total"), decreasing = TRUE)[1:30]
         Demi Lovato            Fifth Harmony               Bebe Rexha               Little Mix                 Jessie J                 Rita Ora
         0.004347826              0.004291845              0.004132231              0.004098361              0.004048583              0.003906250
        Alessia Cara               Katy Perry         Hailee Steinfeld                    Kesha               Nick Jonas              Iggy Azalea
         0.003906250              0.003861004              0.003831418              0.003816794              0.003787879              0.003773585
      Camila Cabello                  Zendaya           Jonas Brothers             Zara Larsson           Kelly Clarkson               Anne-Marie
         0.003759398              0.003717472              0.003703704              0.003649635              0.003649635              0.003610108
          Cher Lloyd           Meghan Trainor           Ellie Goulding                  Tove Lo       Christina Aguilera           Britney Spears
         0.003571429              0.003496503              0.003436426              0.003389831              0.003333333              0.003236246
         Miley Cyrus                     DNCE Selena Gomez & The Scene          Julia Michaels                    Mabel                   Fergie
         0.003205128              0.003174603              0.003144654              0.003125000              0.003095975              0.003095975
> |
```

Compare these scores to the scores for other artists that are related to your artist/band.

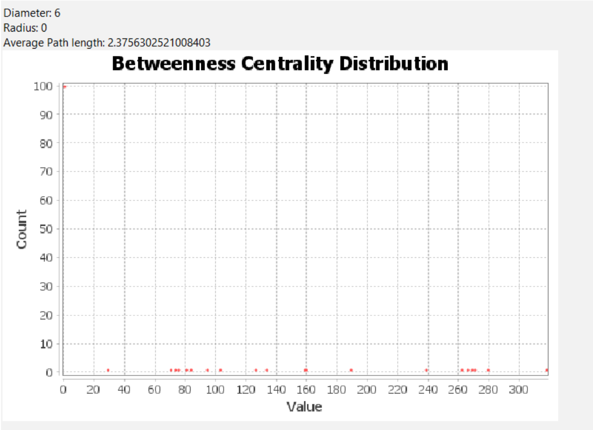
Fifth Harmony:Total degree: 34 (higher than Meghan Trainor's 8)

Total closeness: 0.004291845 (higher)

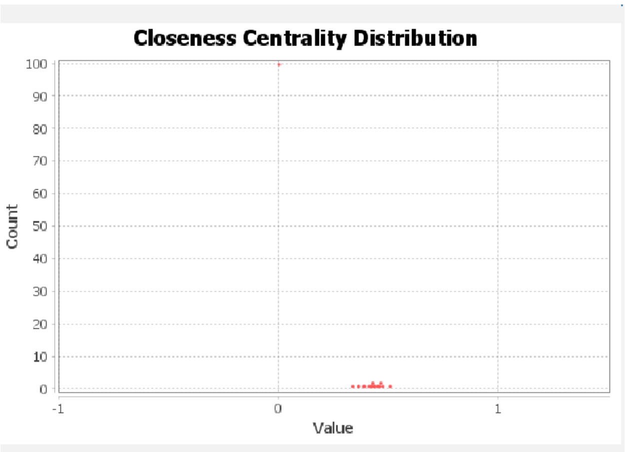
Betweenness: 853.73 (likely higher than 6.61)

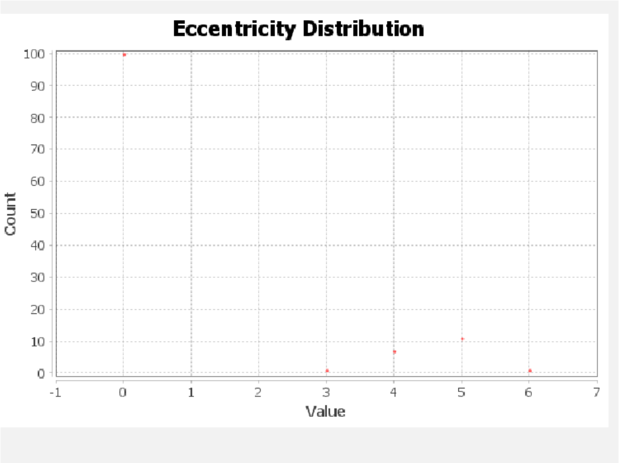


Above is the Closeness Centrality graph from Gephi.



Diameter: 6
Radius: 0
Average Path length: 2.3756302521008403

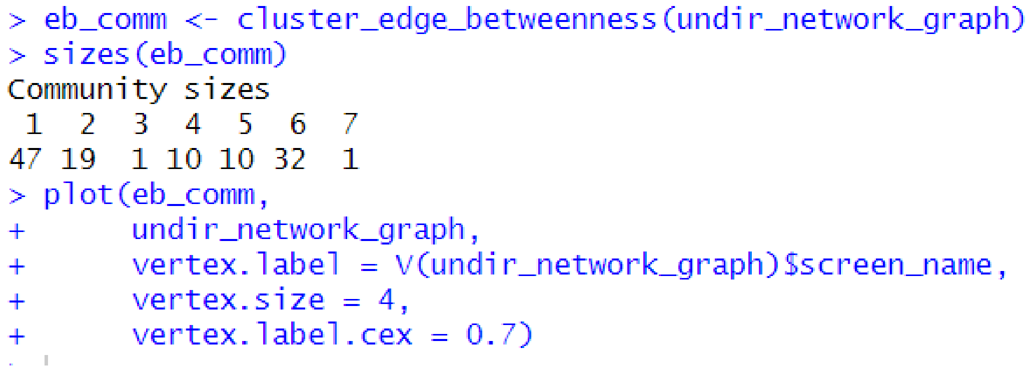




4.2 Community analysis with Girvan-Newman / Louvain methods

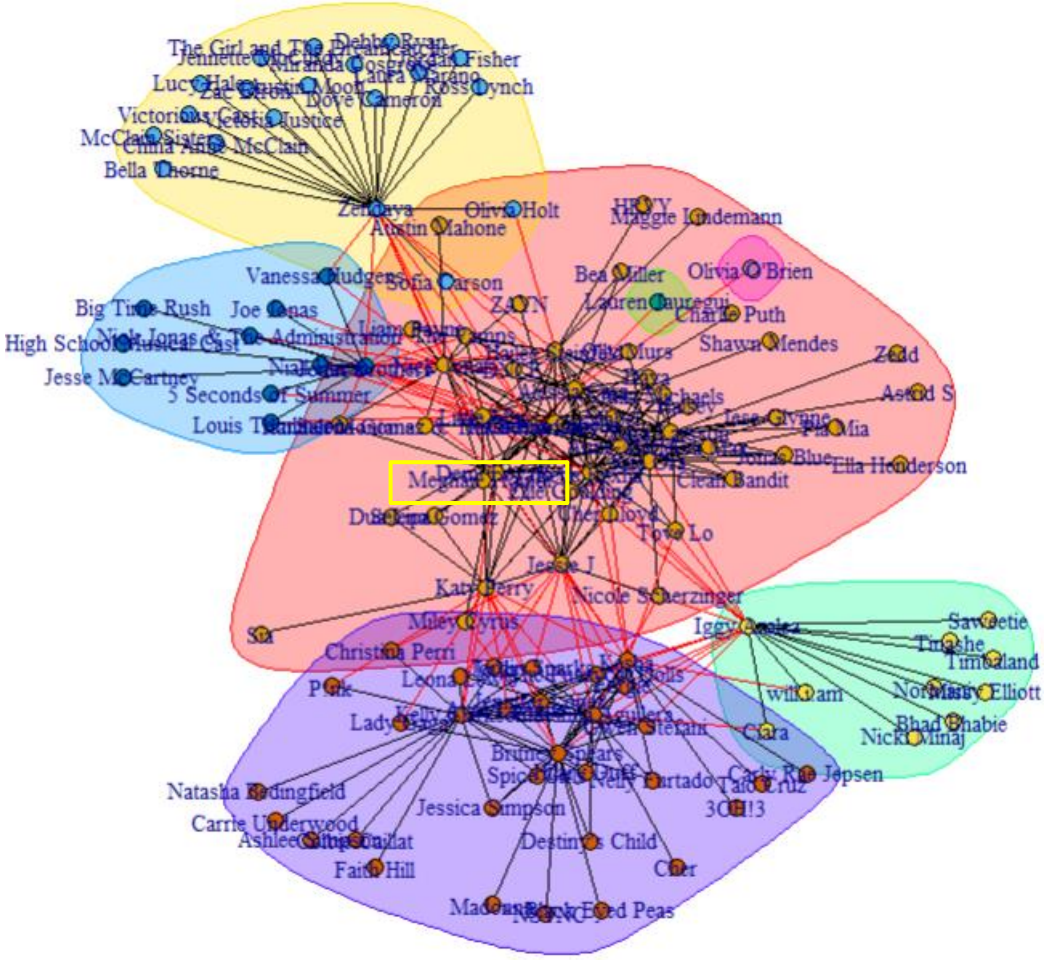
Perform community analysis with the Girvan-Newman (edge betweenness) and Louvain methods.

Girvan-Newman (edge betweenness)

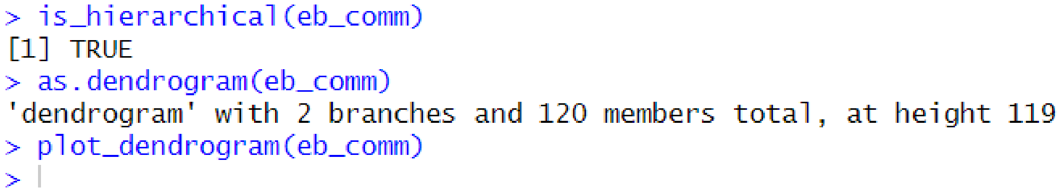


```
> eb_comm <- cluster_edge_betweenness(undir_network_graph)
> sizes(eb_comm)
Community sizes
 1  2  3  4  5  6  7 
47 19  1 10 10 32  1 
> plot(eb_comm,
+      undir_network_graph,
+      vertex.label = V(undir_network_graph)$screen_name,
+      vertex.size = 4,
+      vertex.label.cex = 0.7)
```

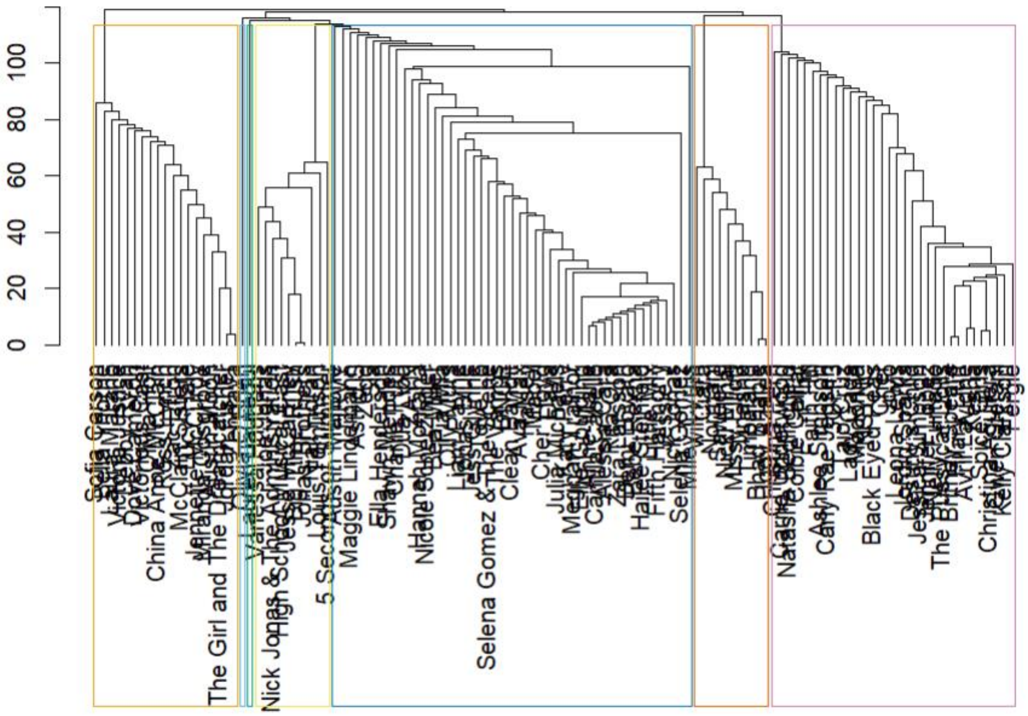
Explain how relevant the results are to your artist/band.



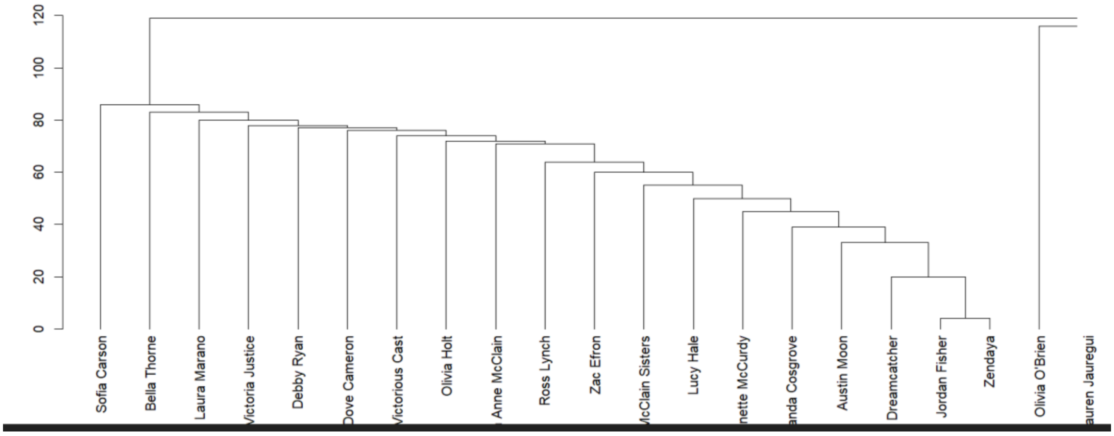
Meghan Trainor is at the center of a dense, large community, show the strength of her influence.



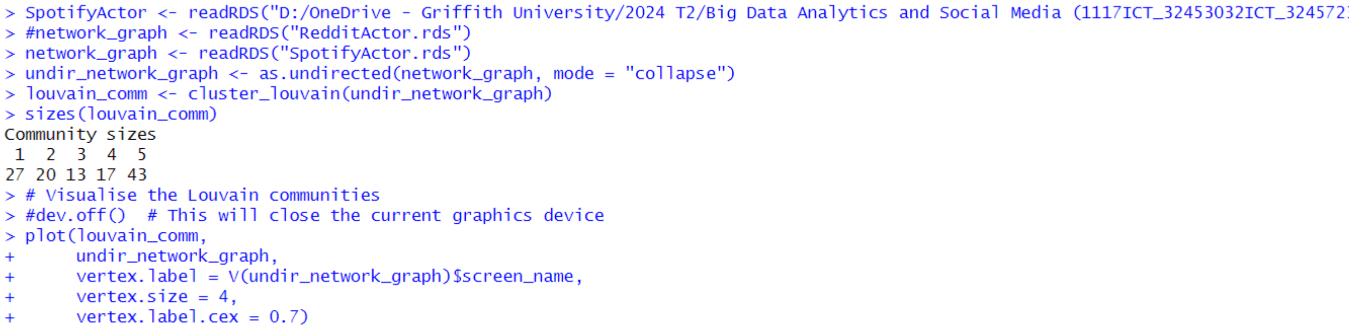
```
> is_hierarchical(eb_comm)
[1] TRUE
> as.dendrogram(eb_comm)
'dendrogram' with 2 branches and 120 members total, at height 119
> plot_dendrogram(eb_comm)
> 
```



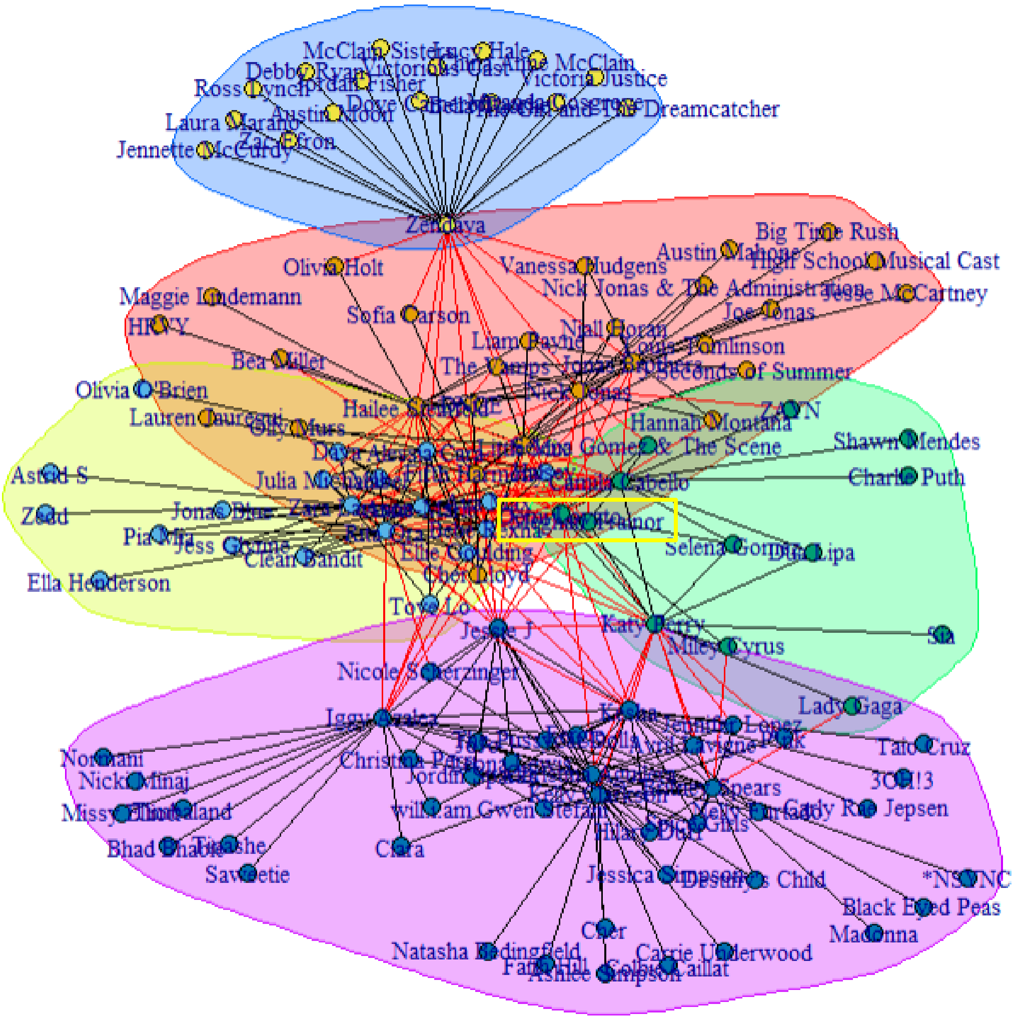
> plot_dendrogram(eb_comm, mode = "dendrogram", xlim = c(1,20))

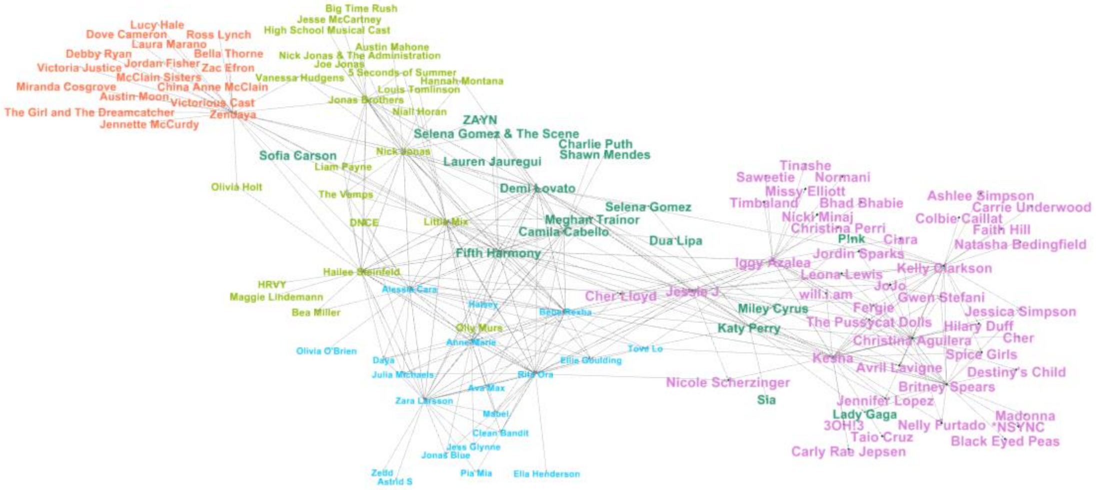


Louvain



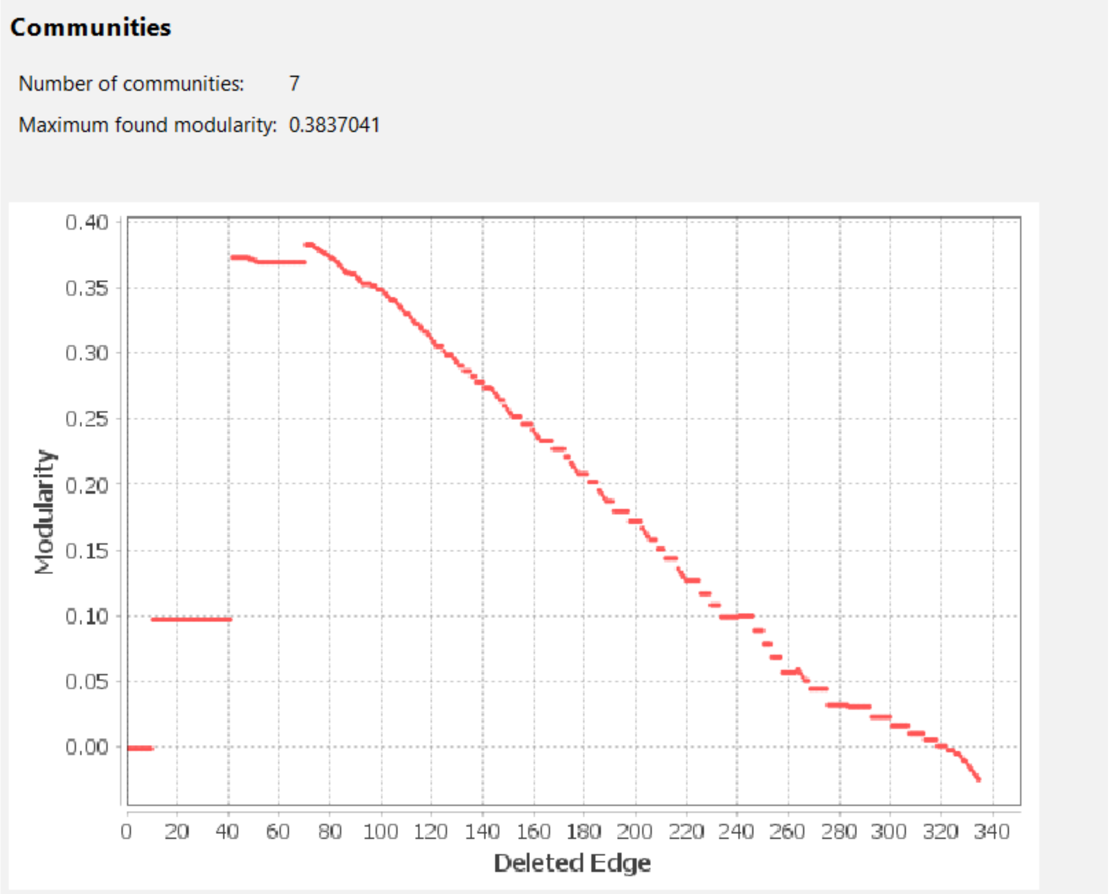
```
> SpotifyActor <- readRDS("D:/OneDrive - Griffith University/2024 T2/Big Data Analytics and Social Media (1117ICT_32453032ICT_324572
> #network_graph <- readRDS("RedditActor.rds")
> network_graph <- readRDS("SpotifyActor.rds")
> undir_network_graph <- as.undirected(network_graph, mode = "collapse")
> louvain_comm <- cluster_louvain(undir_network_graph)
> sizes(louvain_comm)
Community sizes
 1  2  3  4  5 
27 20 13 17 43 
> # Visualise the Louvain communities
> #dev.off()  # This will close the current graphics device
> plot(louvain_comm,
+      undir_network_graph,
+      vertex.label = V(undir_network_graph)$screen_name,
+      vertex.size = 4,
+      vertex.label.cex = 0.7)
```





Modularity Class graph from Gephi

Girvan-Newman Report



Communities

Number of communities: 7

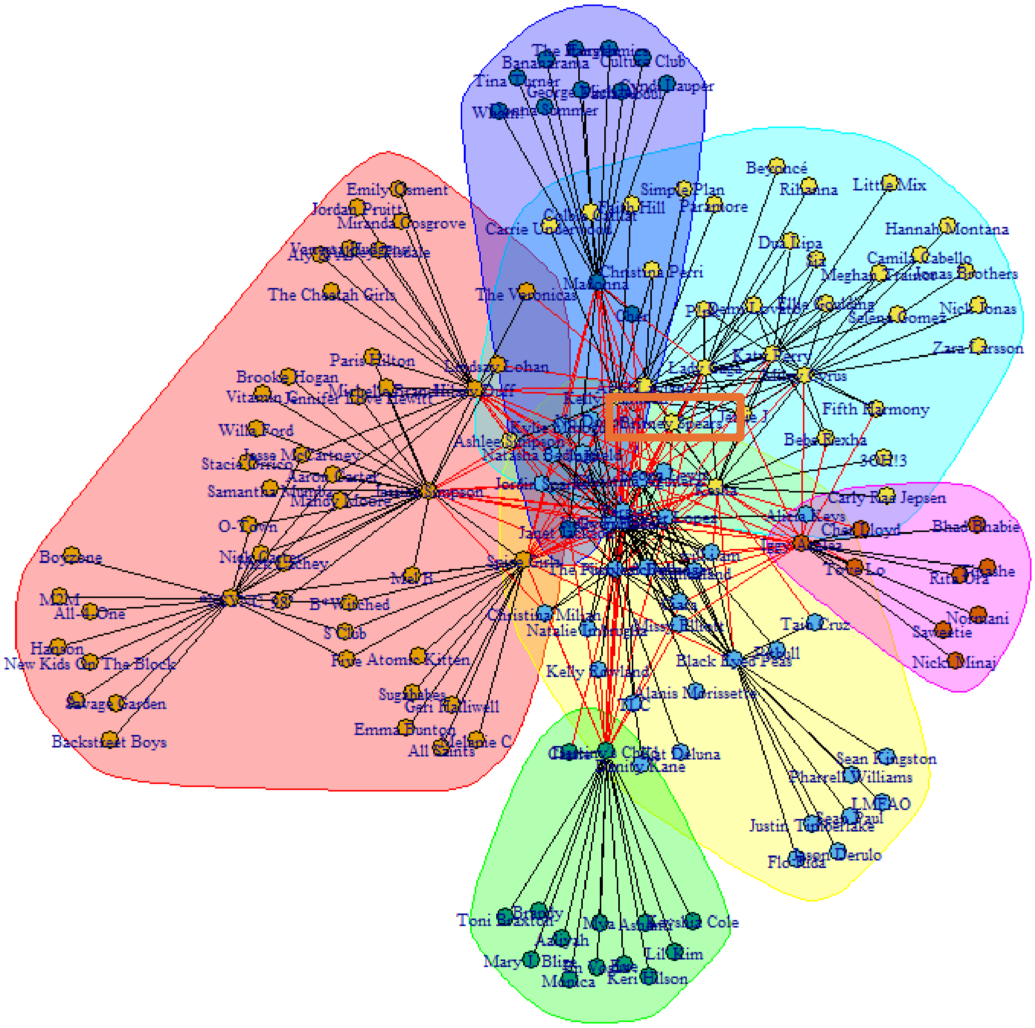
Maximum found modularity: 0.3837041

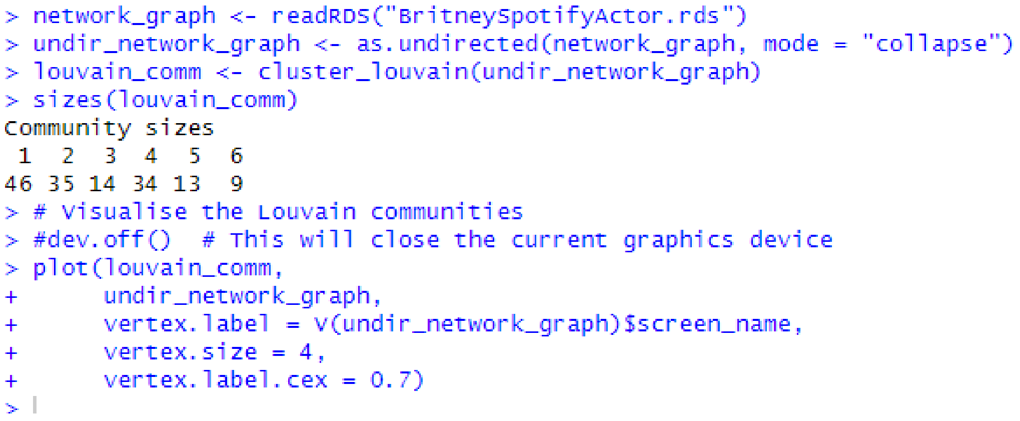
Explain how relevant the results are to your artist/band.

Central position in both plots means Meghan Trainor is not only a key influencer but also a connector within multiple communities. Whether in discussions or artist networks, her presence is significant in shaping conversations or interactions.

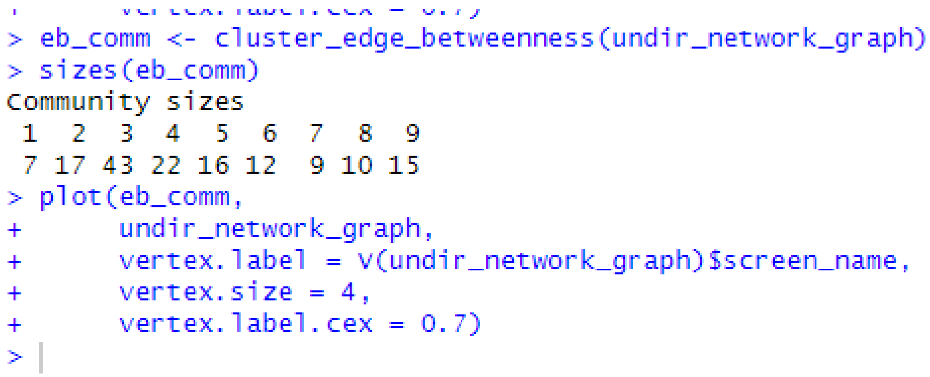
Perform the community analysis also for related artists. Is their community structure similar?

For Britney Spears:

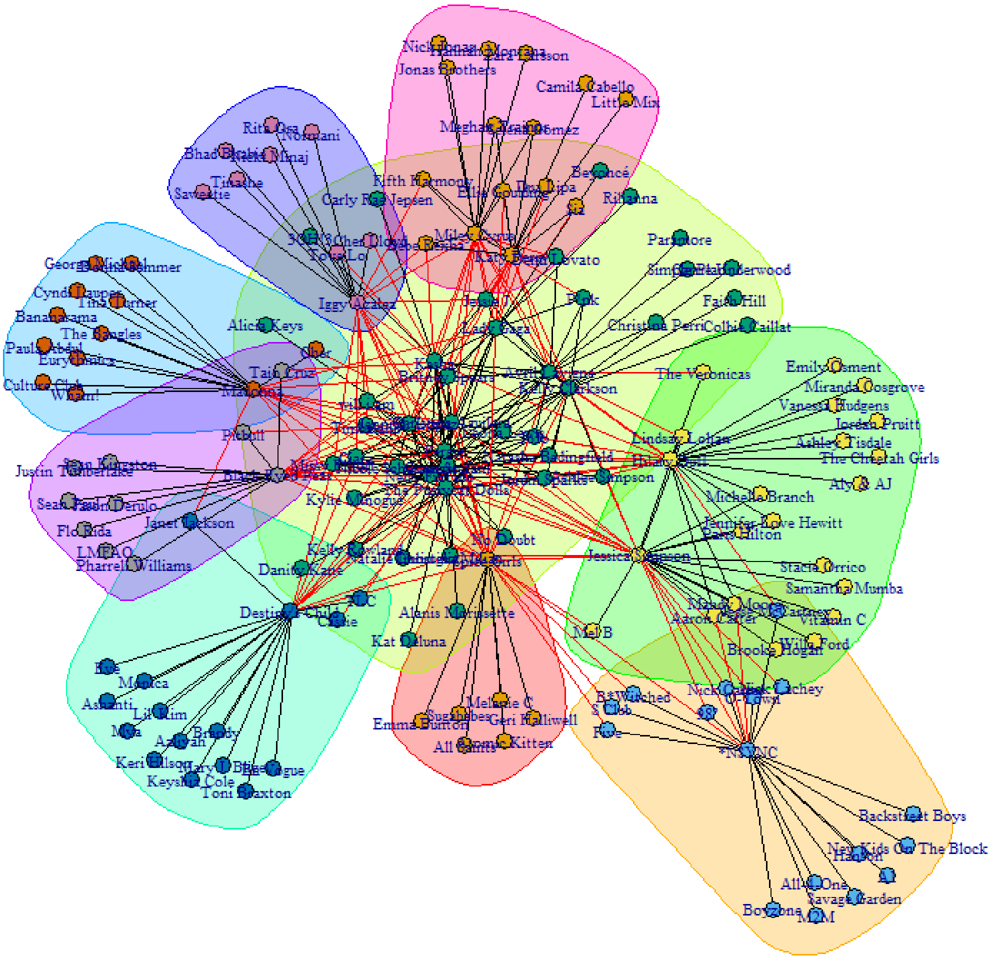


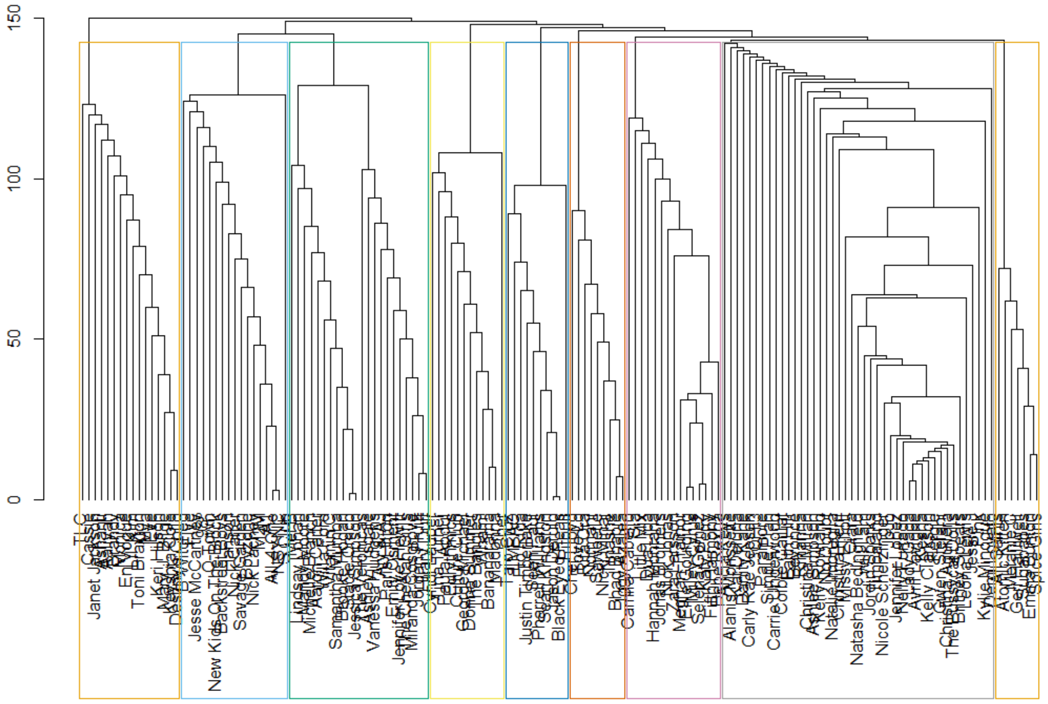


```
> network_graph <- readRDS("BritneySpotifyActor.rds")
> undir_network_graph <- as.undirected(network_graph, mode = "collapse")
> louvain_comm <- cluster_louvain(undir_network_graph)
> sizes(louvain_comm)
Community sizes
 1  2  3  4  5  6 
46 35 14 34 13  9 
> # Visualise the Louvain communities
> #dev.off()  # This will close the current graphics device
> plot(louvain_comm,
+      undir_network_graph,
+      vertex.label = V(undir_network_graph)$screen_name,
+      vertex.size = 4,
+      vertex.label.cex = 0.7)
> 
```



```
+      vertex.label.cex = 0.7)
> eb_comm <- cluster_edge_betweenness(undir_network_graph)
> sizes(eb_comm)
Community sizes
 1  2  3  4  5  6  7  8  9 
 7 17 43 22 16 12  9 10 15 
> plot(eb_comm,
+      undir_network_graph,
+      vertex.label = V(undir_network_graph)$screen_name,
+      vertex.size = 4,
+      vertex.label.cex = 0.7)
> 
```





In Britney's network, the communities seem to be more distinct and separate, while in Meghan's network, the communities overlap more, indicating stronger or more frequent interactions across different groups.

This suggests that Britney's collaborators or associated artists may form tighter, more isolated clusters, possibly based on genre or time periods. In contrast, Meghan's network could indicate a more interconnected web, where artists or collaborators have broader, cross-community interactions.

5. Machine Learning Models

5.1 Sentiment Analysis on public reactions

Use sentiment analysis (5.2) to identify how the public reacts to events and/or topics related to your artist/band. Provide a summary of public opinions (emotions, reactions). [2 marks]

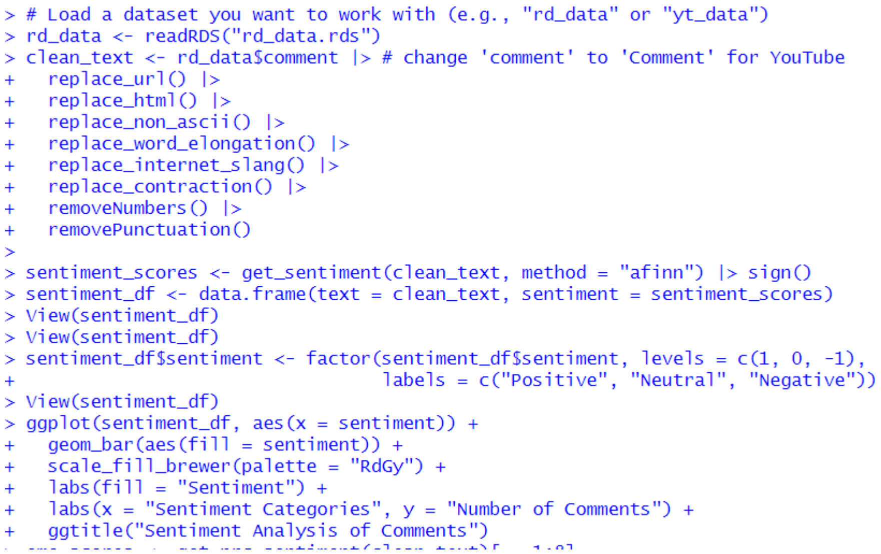
This topic revolves around Meghan Trainor's controversial statement, "Fck teachers," made during a video interview. After running sentiment analysis on the comments, I noticed that while the analysis detected many critical or negative reactions, I believe some of the positive classifications are inaccurate.

For example, the results indicate that emotions like trust (40.8%) and joy (33.9%) were strongly present, which feels misleading given the overall tone of the discussion. Many comments may contain words or phrases that are technically positive, but in reality, they express sarcasm or criticism toward Meghan Trainor's statement. Sentiment analysis seems to struggle in these cases, failing to capture the true intent behind these comments.

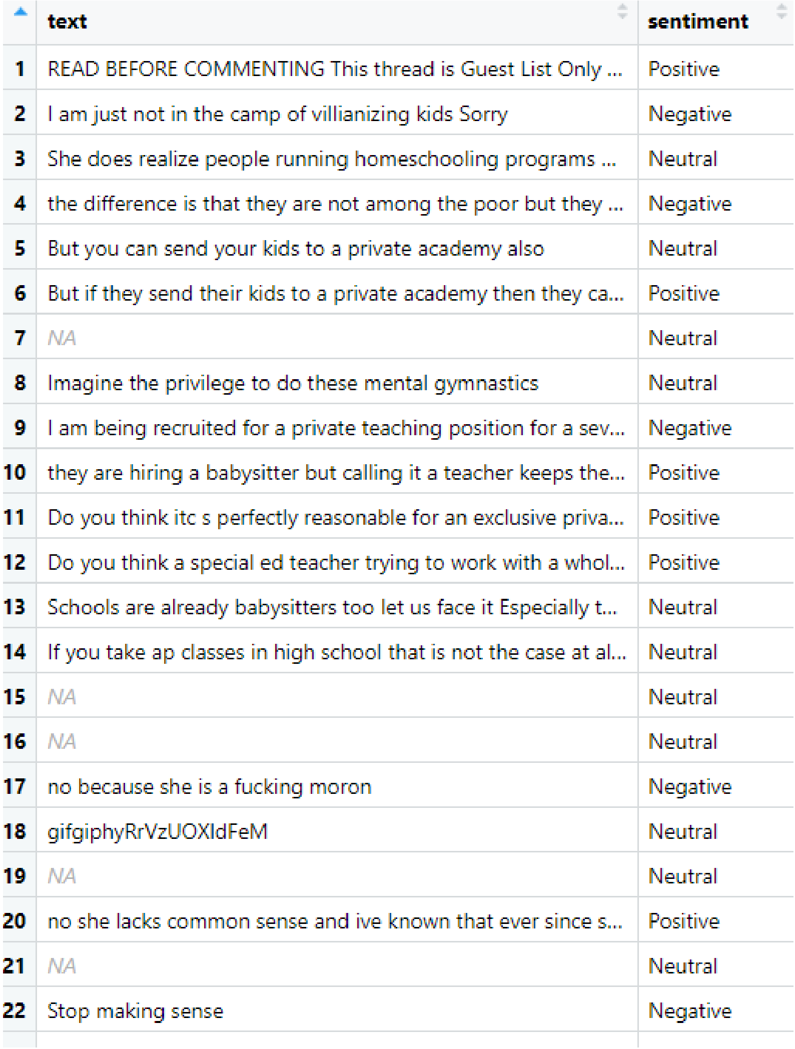
Additionally, while the analysis showed a significant proportion of anticipation (31.9%), sadness (31.9%), and anger (30.2%), these categories seem more aligned with the real emotions in the discussion. However, the presence of trust and joy in such high proportions, despite the negative context, suggests that the tool's assessment may be oversimplified.

The comment breakdown also supports this view. While some statements were detected as neutral or positive, such as, "If they send their kids to a private academy then they can..." (classified as positive), I think that these assessments miss the underlying sarcasm and criticism directed at Meghan’s view. The analysis marks these as positive, but I believe they're opposite statements meant to disagree with her position.

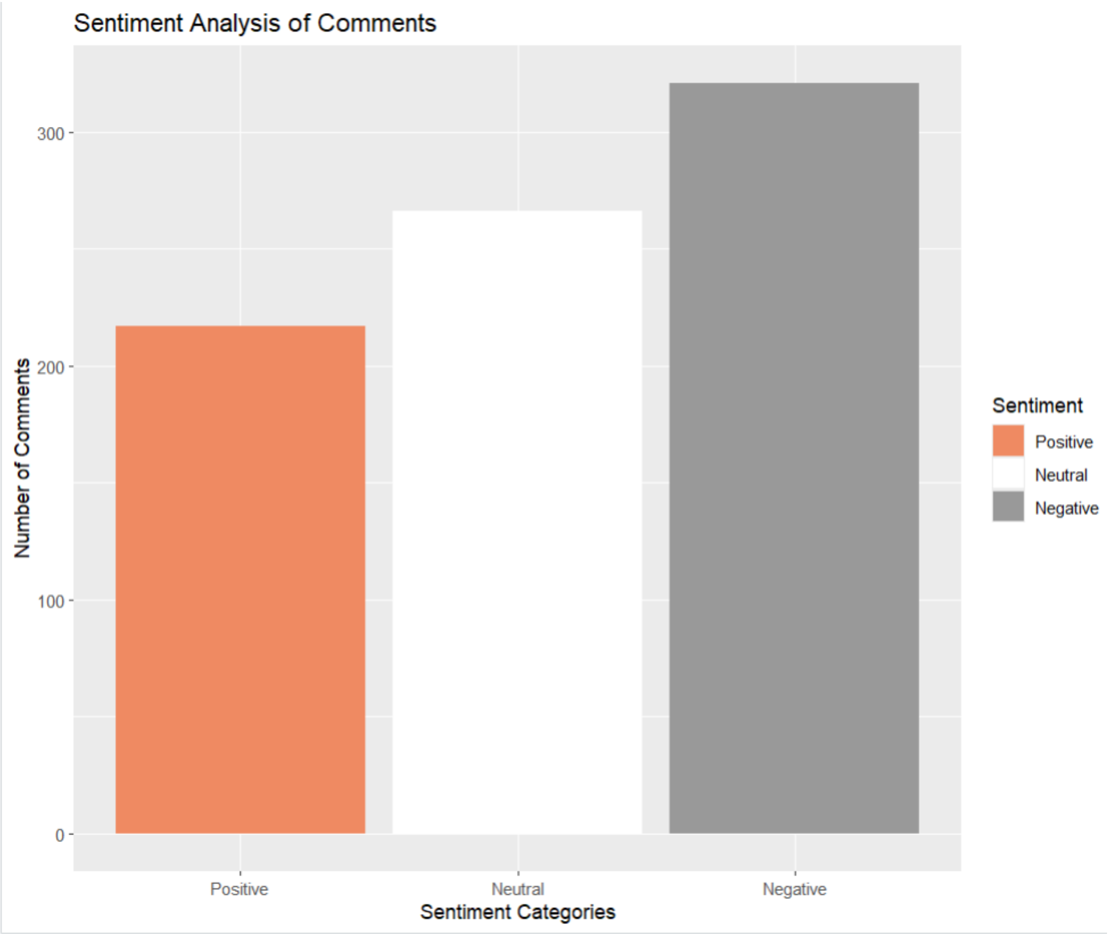
In summary, while sentiment analysis is helpful in gauging overall reactions, it struggles to detect sarcasm and complex emotions, leading to overestimation of positive categories like trust and joy in a discussion that I personally interpret as much more negative.

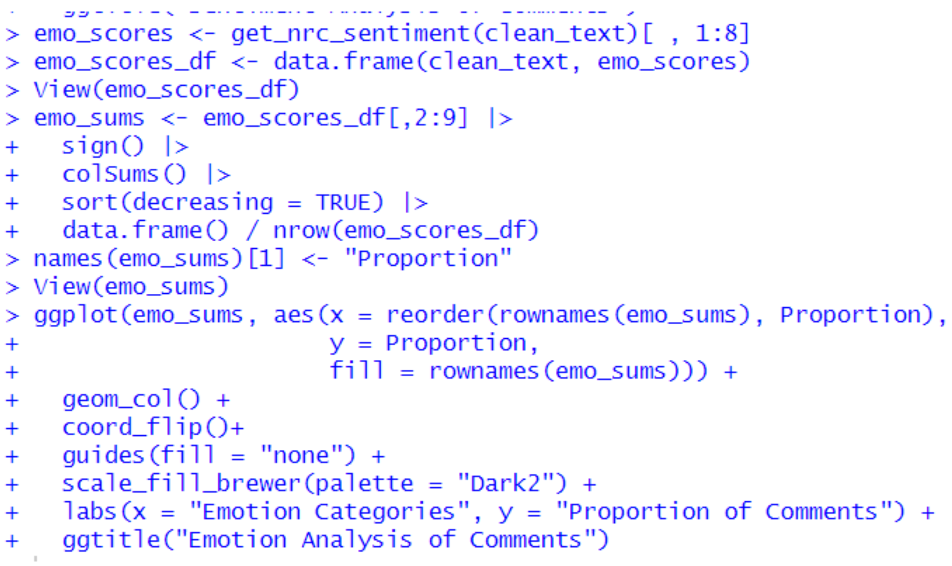


```
> # Load a dataset you want to work with (e.g., "rd_data" or "yt_data")
> rd_data <- readRDS("rd_data.rds")
> clean_text <- rd_data$comment |> # change 'comment' to 'Comment' for YouTube
+   replace_url() |>
+   replace_html() |>
+   replace_non_ascii() |>
+   replace_word_elongation() |>
+   replace_internet_slang() |>
+   replace_contraction() |>
+   removeNumbers() |>
+   removePunctuation()
> 
> sentiment_scores <- get_sentiment(clean_text, method = "afinn") |> sign()
> sentiment_df <- data.frame(text = clean_text, sentiment = sentiment_scores)
> View(sentiment_df)
> View(sentiment_df)
> sentiment_df$sentiment <- factor(sentiment_df$sentiment, levels = c(1, 0, -1),
+                                  labels = c("Positive", "Neutral", "Negative"))
> View(sentiment_df)
> ggplot(sentiment_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")
> emo_scores <- get_nrc_sentiment(clean_text)[ , 1:8]
```

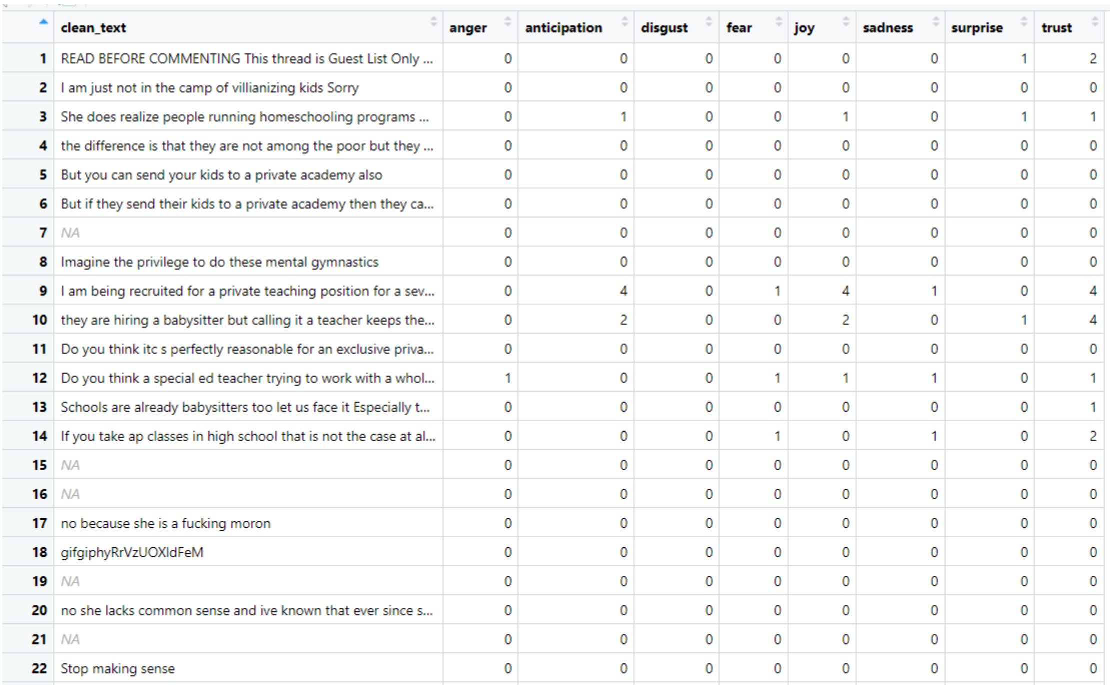


	text	sentiment
1	READ BEFORE COMMENTING This thread is Guest List Only ...	Positive
2	I am just not in the camp of villianizing kids Sorry	Negative
3	She does realize people running homeschooling programs ...	Neutral
4	the difference is that they are not among the poor but they ...	Negative
5	But you can send your kids to a private academy also	Neutral
6	But if they send their kids to a private academy then they ca...	Positive
7	NA	Neutral
8	Imagine the privilege to do these mental gymnastics	Neutral
9	I am being recruited for a private teaching position for a sev...	Negative
10	they are hiring a babysitter but calling it a teacher keeps the...	Positive
11	Do you think itc s perfectly reasonable for an exclusive priva...	Positive
12	Do you think a special ed teacher trying to work with a whol...	Positive
13	Schools are already babysitters too let us face it Especially t...	Neutral
14	If you take ap classes in high school that is not the case at al...	Neutral
15	NA	Neutral
16	NA	Neutral
17	no because she is a fucking moron	Negative
18	gifgiphyRrVzUOXIdFeM	Neutral
19	NA	Neutral
20	no she lacks common sense and ive known that ever since s...	Positive
21	NA	Neutral
22	Stop making sense	Negative

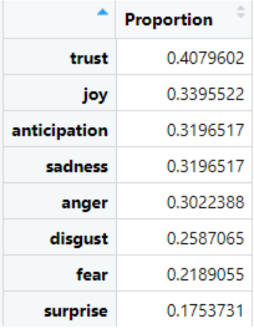




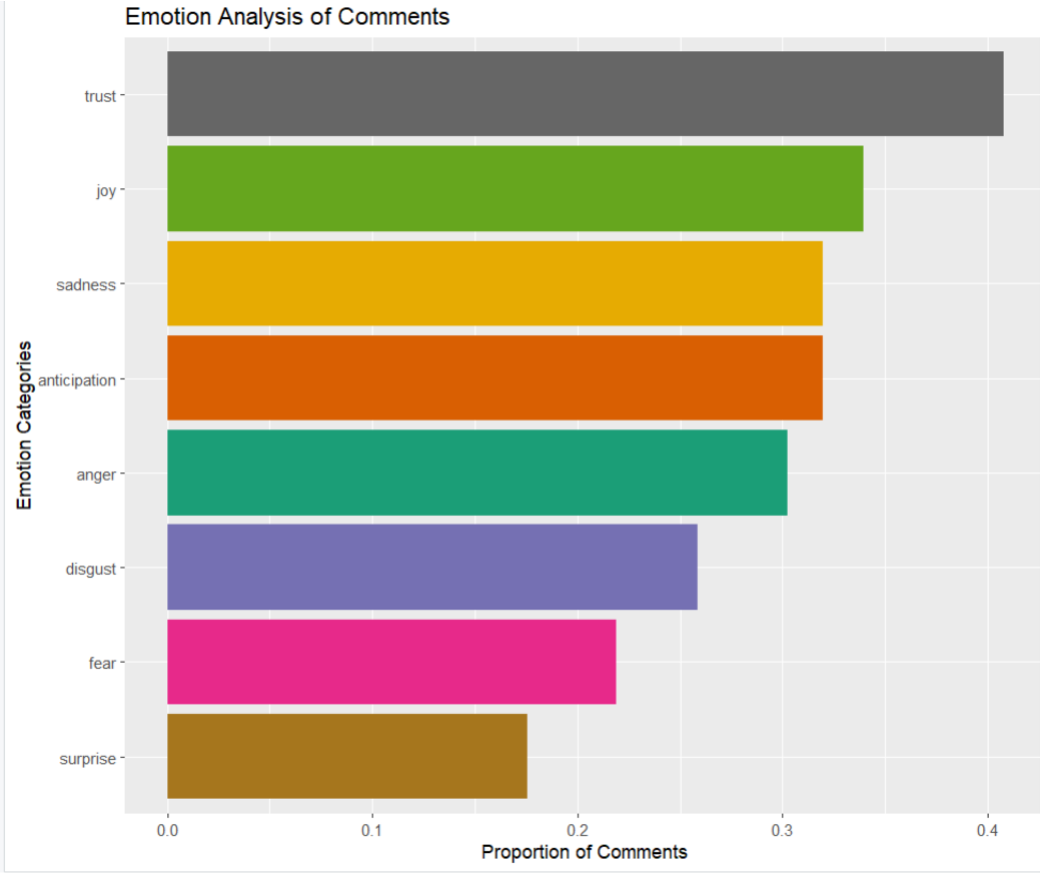
```
+   ggtitle("Sentiment Analysis of Comments")
> emo_scores <- get_nrc_sentiment(clean_text)[ , 1:8]
> emo_scores_df <- data.frame(clean_text, emo_scores)
> View(emo_scores_df)
> emo_sums <- emo_scores_df[,2:9] |>
+   sign() |>
+   colSums() |>
+   sort(decreasing = TRUE) |>
+   data.frame() / nrow(emo_scores_df)
> names(emo_sums)[1] <- "Proportion"
> View(emo_sums)
> ggplot(emo_sums, aes(x = reorder(rownames(emo_sums), Proportion),
+                      y = Proportion,
+                      fill = rownames(emo_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")
```



	clean_text	anger	anticipation	disgust	fear	joy	sadness	surprise	trust
1	READ BEFORE COMMENTING This thread is Guest List Only ...	0	0	0	0	0	0	1	2
2	I am just not in the camp of villianizing kids Sorry	0	0	0	0	0	0	0	0
3	She does realize people running homeschooling programs ...	0	1	0	0	1	0	1	1
4	the difference is that they are not among the poor but they ...	0	0	0	0	0	0	0	0
5	But you can send your kids to a private academy also	0	0	0	0	0	0	0	0
6	But if they send their kids to a private academy then they ca...	0	0	0	0	0	0	0	0
7	NA	0	0	0	0	0	0	0	0
8	Imagine the privilege to do these mental gymnastics	0	0	0	0	0	0	0	0
9	I am being recruited for a private teaching position for a sev...	0	4	0	1	4	1	0	4
10	they are hiring a babysitter but calling it a teacher keeps the...	0	2	0	0	2	0	1	4
11	Do you think itc s perfectly reasonable for an exclusive priva...	0	0	0	0	0	0	0	0
12	Do you think a special ed teacher trying to work with a whol...	1	0	0	1	1	1	0	1
13	Schools are already babysitters too let us face it Especially t...	0	0	0	0	0	0	0	1
14	If you take ap classes in high school that is not the case at al...	0	0	0	1	0	1	0	2
15	NA	0	0	0	0	0	0	0	0
16	NA	0	0	0	0	0	0	0	0
17	no because she is a fucking moron	0	0	0	0	0	0	0	0
18	gifgiphyRrVzUOXIdFeM	0	0	0	0	0	0	0	0
19	NA	0	0	0	0	0	0	0	0
20	no she lacks common sense and ive known that ever since s...	0	0	0	0	0	0	0	0
21	NA	0	0	0	0	0	0	0	0
22	Stop making sense	0	0	0	0	0	0	0	0



	Proportion
trust	0.4079602
joy	0.3395522
anticipation	0.3196517
sadness	0.3196517
anger	0.3022388
disgust	0.2587065
fear	0.2189055
surprise	0.1753731



5.2 Decision Tree performance Prediction

Build a decision tree(5.2) and evaluate its performance in predicting whether a song is by your artist/band.

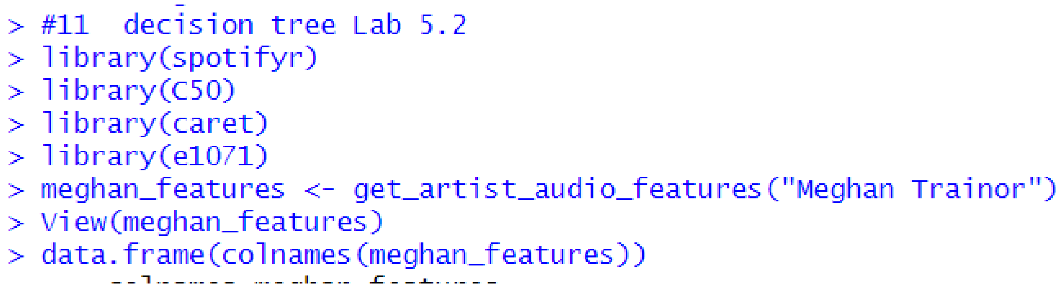
```
[1] "Prediction is: 1. Correct!"
        Reference
Prediction    0    1
        0  5.7  4.7
        1 13.4 76.3

 Accuracy (average) : 0.8196
```

I built a decision tree using the C5.0 algorithm to predict whether a song is by Meghan Trainor. The model was trained on a combined dataset of audio features from Meghan Trainor's songs and a top 50 Spotify playlist. After splitting the data (80% training, 20% testing), the model achieved an accuracy of 83.75% in predicting whether a song was by Meghan Trainor.

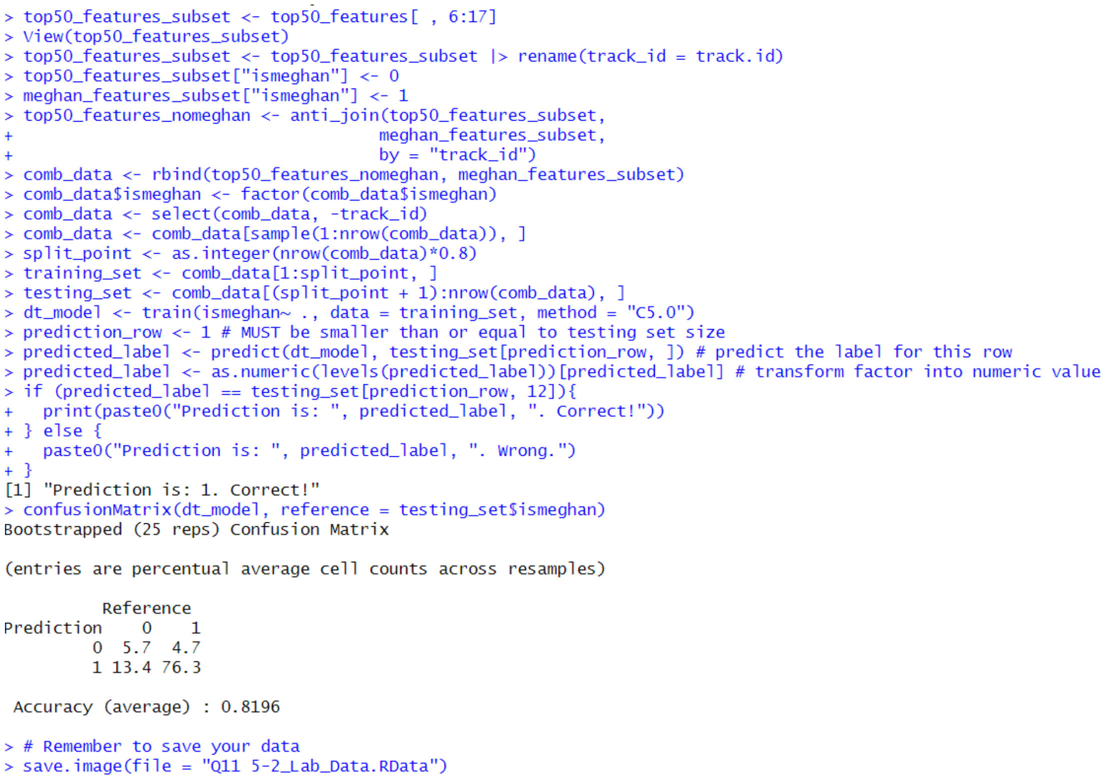
79.7% accuracy for Meghan Trainor's songs comes from the matrix showing that when the model predicted "Meghan Trainor," 79.7% of those predictions were correct.

13.2% false positive rate means that 13.2% of the songs were predicted to be by Meghan Trainor, but they were actually by other artists.

The overall 83.75% accuracy is the average percentage of correct predictions across the entire test set.



```
> #11  decision tree Lab 5.2
> library(spotifyr)
> library(C50)
> library(caret)
> library(e1071)
> meghan_features <- get_artist_audio_features("Meghan Trainor")
> View(meghan_features)
> data.frame(colnames(meghan_features))
```



```
> top50_features_subset <- top50_features[ , 6:17]
> View(top50_features_subset)
> top50_features_subset <- top50_features_subset |> rename(track_id = track.id)
> top50_features_subset["ismeghan"] <- 0
> meghan_features_subset["ismeghan"] <- 1
> top50_features_nomeghan <- anti_join(top50_features_subset,
+                                     meghan_features_subset,
+                                     by = "track_id")
> comb_data <- rbind(top50_features_nomeghan, meghan_features_subset)
> comb_data$ismeghan <- factor(comb_data$ismeghan)
> comb_data <- select(comb_data, -track_id)
> comb_data <- comb_data[sample(1:nrow(comb_data)), ]
> 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), ]
> dt_model <- train(ismeghan~ ., data = training_set, method = "C5.0")
> prediction_row <- 1 # MUST be smaller than or equal to testing set size
> predicted_label <- predict(dt_model, testing_set[prediction_row, ]) # predict the label for this row
> predicted_label <- as.numeric(levels(predicted_label))[predicted_label] # transform factor into numeric value
> if (predicted_label == testing_set[prediction_row, 12]){
+   print(paste0("Prediction is: ", predicted_label, ". Correct!"))
+ } else {
+   paste0("Prediction is: ", predicted_label, ". Wrong.")
+ }
[1] "Prediction is: 1. Correct!"
> confusionMatrix(dt_model, reference = testing_set$ismeghan)
Bootstrapped (25 reps) Confusion Matrix 

(entries are percentual average cell counts across resamples)
 
          Reference
Prediction    0    1
         0  5.7  4.7
         1 13.4 76.3
                            
 Accuracy (average) : 0.8196

> # Remember to save your data
> save.image(file = "Q11 5-2_Lab_Data.RData")
```

5.3 LDA Topic Modelling to identify related terms

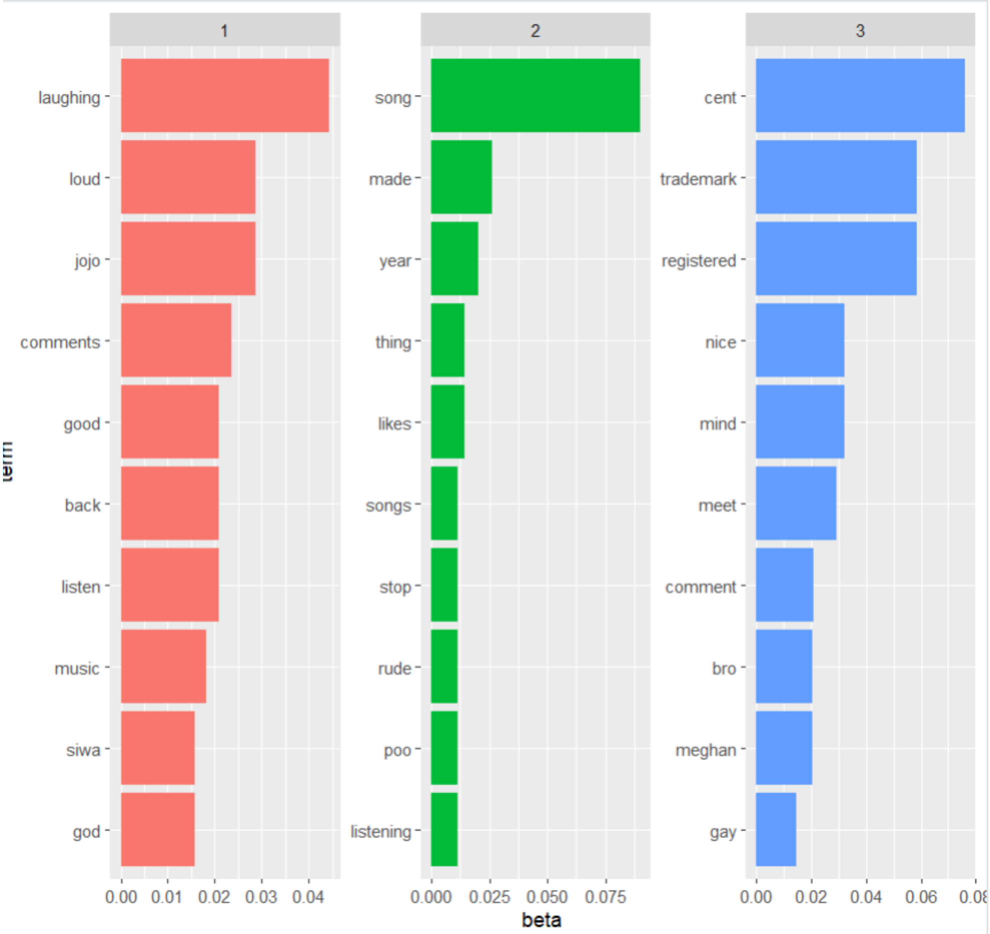
Use LDA topic modelling to identify some terms that are closely related to your artist/band. Find at least 3 significant groups of words that can be meaningful to your analysis. Explain your findings. [2 marks]

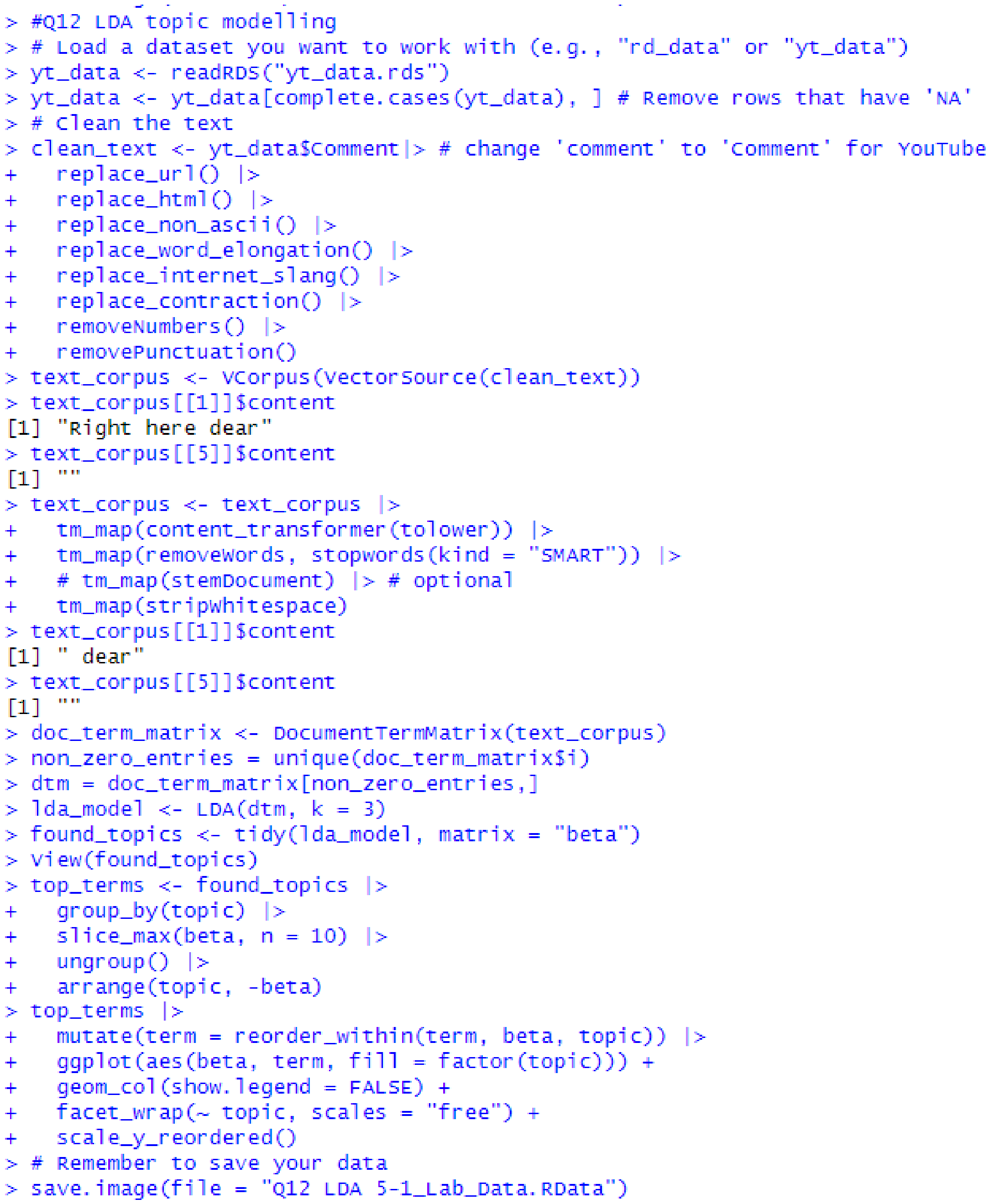
In the LDA topic modeling of the YouTube comments for Meghan Trainor's "Make You Look" video, three key topics emerged:

Topic 1 (Red Bar): Focuses on emotional reactions, with terms like "laughing," "loud," and "comments," showing people’s humorous responses, possibly related to her interactions with JoJo Siwa.

Topic 2 (Green Bar): Centers on her music, including terms like "song," "made," and "likes," indicating discussions about her songs and their impact.

Topic 3 (Blue Bar): Highlights her branding and identity, with terms like "trademark," "registered," and "gay," reflecting her public image and possible LGBTQ+ support.



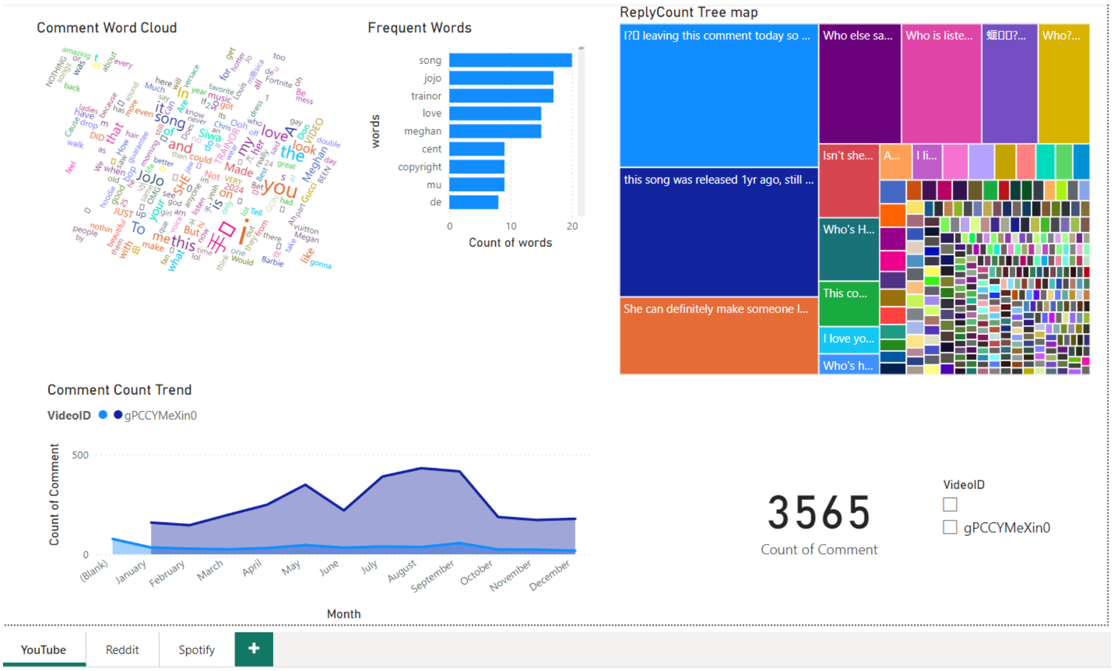


```
> #Q12 LDA topic modelling
> # Load a dataset you want to work with (e.g., "rd_data" or "yt_data")
> yt_data <- readRDS("yt_data.rds")
> yt_data <- yt_data[complete.cases(yt_data), ] # Remove rows that have 'NA'
> # Clean the text
> clean_text <- yt_data$Comment|> # change 'comment' to 'Comment' for YouTube
+   replace_url() |>
+   replace_html() |>
+   replace_non_ascii() |>
+   replace_word_elongation() |>
+   replace_internet_slang() |>
+   replace_contraction() |>
+   removeNumbers() |>
+   removePunctuation()
> text_corpus <- VCorpus(VectorSource(clean_text))
> text_corpus[[1]]$content
[1] "Right here dear"
> text_corpus[[5]]$content
[1] ""
> text_corpus <- text_corpus |>
+   tm_map(content_transformer(tolower)) |>
+   tm_map(removeWords, stopwords(kind = "SMART")) |>
+   # tm_map(stemDocument) |> # optional
+   tm_map(stripWhitespace)
> text_corpus[[1]]$content
[1] " dear"
> text_corpus[[5]]$content
[1] ""
> doc_term_matrix <- DocumentTermMatrix(text_corpus)
> non_zero_entries = unique(doc_term_matrix$i)
> dtm = doc_term_matrix[non_zero_entries,]
> lda_model <- LDA(dtm, k = 3)
> found_topics <- tidy(lda_model, matrix = "beta")
> View(found_topics)
> top_terms <- found_topics |>
+   group_by(topic) |>
+   slice_max(beta, n = 10) |>
+   ungroup() |>
+   arrange(topic, -beta)
> top_terms |>
+   mutate(term = reorder_within(term, beta, topic)) |>
+   ggplot(aes(beta, term, fill = factor(topic))) +
+   geom_col(show.legend = FALSE) +
+   facet_wrap(~ topic, scales = "free") +
+   scale_y_reordered()
> # Remember to save your data
> save.image(file = "Q12 LDA 5-1_Lab_Data.RData")
```

6. Power BI Visualisation

Create two dashboards(4.2) (pages), each with at least three charts, from your datasets using Power BI. Describe each chart in your dashboard and why you chose to include it. Explain the functionality of your dashboard and what insights you can obtain from it. [3 marks] Analysis Review

6.1 YouTube Dashboard

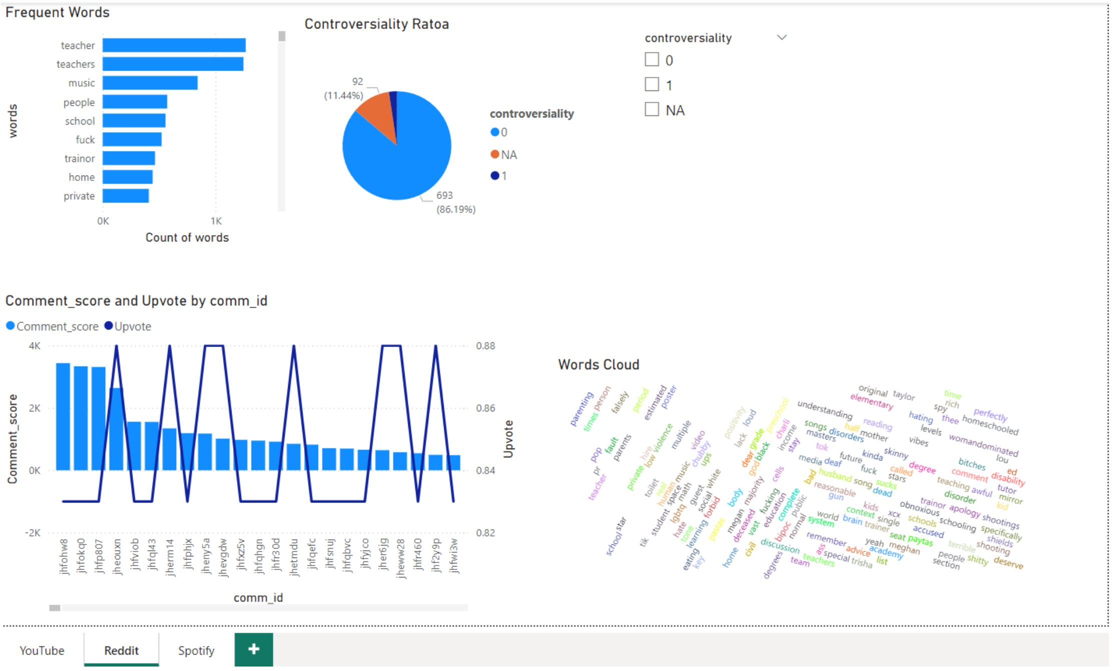


YouTube Dashboard:

- Comment Word Cloud: This chart displays the most frequently used words from YouTube comments on Meghan Trainor's video. It provides an immediate overview of key topics that people are discussing. It was chosen to quickly identify trending words and themes that stood out in the comments.
- Frequent Words Bar Chart: This bar chart gives a more structured representation of the most frequent words, making it easier to focus on specific terms and their frequency. The most frequent two words is “song” and “jojo”.
- ReplyCount Tree Map: This treemap visualizes which comments have received the most replies, providing insight into the conversations generating the most engagement. It helps highlight what sparked deeper discussions or debates.
- Comment Count Trend: This line chart tracks the number of comments over time, helping to identify when interest in the video peaked. It was chosen to understand the time-based engagement trends.
- Total Comment Count: This metric shows the total number of comments,

providing a quick snapshot of overall engagement with the video.

6.2 Reddit Dashboard:

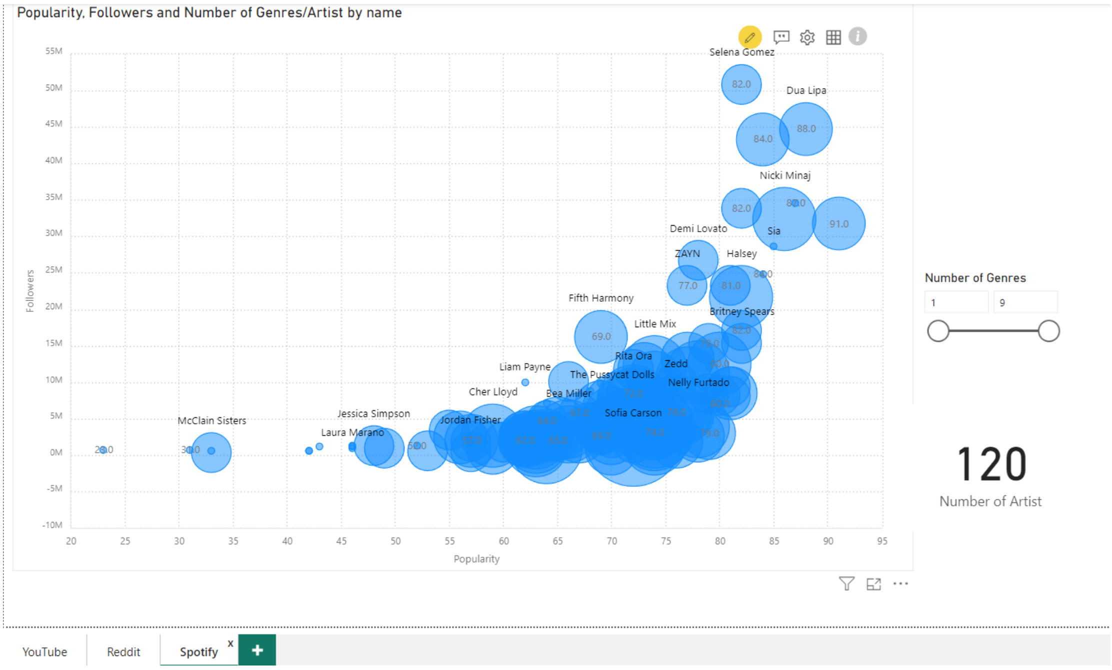


- The Controversiality Ratio Pie Chart reveals that 86.19% of the comments were marked as controversial, highlighting how polarizing Meghan Trainor's comments about teachers were on Reddit. This demonstrates that the discussion sparked strong reactions, with a large portion of the audience engaging in contentious debates.

- Comment Score and Upvote Chart: This dual-axis chart compares comment scores with upvotes, allowing us to see how the community valued different comments. It was included to evaluate the popularity and approval of specific contributions.

- Word Cloud: A word cloud showing the most frequent words used in Reddit discussions. Similar to the YouTube word cloud, it helps visualize the major topics being discussed in a more unstructured format.

6.2 Spotify Dashboard:



Popularity, Followers, and Genres Bubble Chart: This bubble chart represents the relationship between artist popularity, their number of followers, and the diversity of genres they cover. It was included to compare Meghan Trainor's standing against other artists in terms of both fanbase and musical diversity. The chart helps to analyze how popularity correlates with the variety of music styles.

Number of Artists: This figure shows the number of artists included in the analysis. It offers insight into the dataset's scope and helps put the comparisons into context.

Dashboard Functionality: In my dashboard, I added functionality through filters such as video ID, controversiality, and number of genres. The controversiality filter helps focus on comments with high engagement or strong opinions. **Video ID** isolates data for a user ID. And **Number of Genres** filter helps analyze content that spans multiple genres.

7. Analysis Review

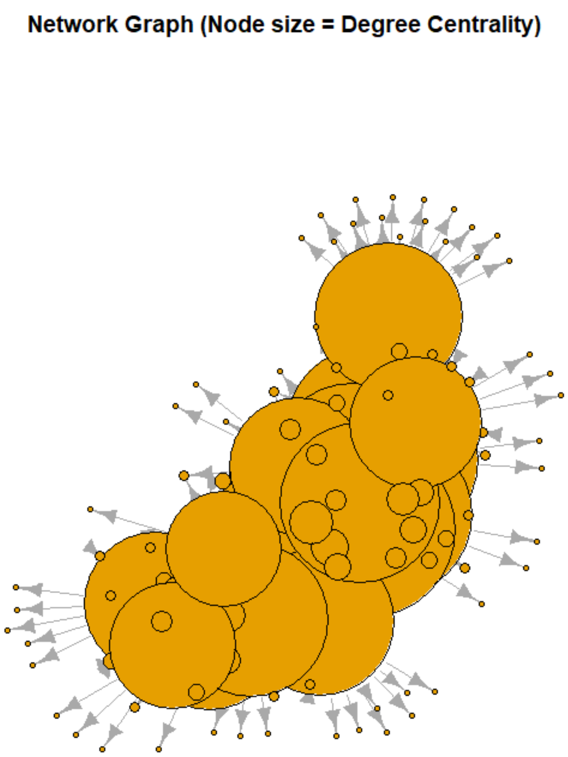
7.1 Eigenvector Centrality network analysis

Research and review other methods/algorithms for network analysis, machine learning models, or visualisation. Compare them to the methods you used in this Assignment. Did you find a method that could give you better insights or more promising results for your social media analytics? Explain why you think so.

I used Eigenvector Centrality to perform a more detailed analysis of the network. This method identifies the most influential nodes by not only considering how many connections each node has but also the importance of those connections. In my case, artists like Bebe Rexha, Fifth Harmony, and Demi Lovato were ranked highest in terms of influence because they are well-connected to other influential nodes.

The difference between Eigenvector Centrality and community detection algorithms like Girvan-Newman and Louvain is that eigenvector centrality focuses on individual node influence, while Girvan-Newman and Louvain focus on finding clusters or communities in the network. The community detection methods helped identify tightly-knit groups of artists, but they did not highlight key influencers as effectively as eigenvector centrality.

Using eigenvector centrality allowed me to gain better insights into which artists are most impactful in the network, whereas the community detection algorithms provided more structural information about the network's groupings.



Network Graph (Node size = Degree Centrality)

```
> #Q14
> # Load required libraries
> library(igraph)
> # Load the network graph (replace with your actual file)
> network_graph <- readRDS("SpotifyActor.rds")
> # Inspect the graph
> summary(network_graph)
IGRAPH 978290d DN-- 120 400 --
+ attr: name (v/c)
> # Find the largest connected component
> comps <- components(network_graph, mode = "weak")
> largest_comp <- which.max(comps$csize)
> comp_subgraph <- induced_subgraph(network_graph, vids = which(comps
$membership == largest_comp))
> # Calculate Eigenvector Centrality
> eigen_centrality <- evcent(comp_subgraph)$vector
> cat("Top 5 nodes by Eigenvector Centrality:\n")
Top 5 nodes by Eigenvector Centrality:
> print(sort(eigen_centrality, decreasing = TRUE)[1:5])
   Bebe Rexha Fifth Harmony  Demi Lovato    Little Mix  Alessia Cara
    1.0000000     0.9997142     0.9090609     0.9011678     0.8842768
```

```
> # Plot the network with eigenvector centrality size scaling
> plot(comp_subgraph, vertex.size = eigen_centrality*20, vertex.label
 = NA, main = "Network Graph (Node size = Eigenvector Centrality)")
```

8. Reference

Wikipedia contributors. (2024b, September 21). *Meghan Trainor*. Wikipedia. https://en.wikipedia.org/wiki/Meghan_Trainor