Machine Learning Social Media Analysis: Drake Case Study

Case Study Setting

Q. 1.1:

Describe the artist/band you are managing. Make sure to reference your sources properly (don't plagiarise). Use APA referencing style. For example: - How many years have they been active? - How many albums & songs have they published? [1-2 paragraphs, 2 marks]

ANS.

The artist chosen by me for this assignment is 'Drake'. Drake hails from Canada. Drake, a key player in modern popular music, is credited with bringing singing and R&B elements to hip hop. (Wikipedia, n.d.). Drake's music career began in 2006 when he released his first mixtape, "Room for Improvement." Three years later, his third mixtape, "So Far Gone," brought him both critical and financial success. The following year, to mostly favourable reviews, he released his formal debut album, "Thank Me Later." (IMDB, n.d.). So, Drake has been an active Musician for 17 years. In the course of 17 years, he has 140 singles (including 81 as a featured artist), five promotional singles, seven studio albums, three compilation albums, four extended plays, seven mixtapes, and 84 music videos. (Wikipedia, n.d.).

Data Selection & Exploration

Q. 1.2: Collect data about your artist/band from Twitter. Make sure to choose keywords for data retrieval that are most relevant to your artist/band. However, try not to be too narrow. As a rough guide, you should retrieve at least 1000 tweets. List the keywords, explain your search strategy, and how much data you have collected. *(=> see Lab 2.1 for help)* [1.5 marks]

ANS.

Drake has earned many nicknames like Drizzy, Champagne Papi, 6 God, Young Angel etc. However, Drizzy and Champagne Papi are most frequently associated with him. He even has his Instagram username as ‘champagnepapi’ & his Twitter account name as ‘Drizzy’. So, I tried searching tweets related to him by using 3 keywords: ‘champagnepapi’, ‘drizzy’ and ‘drake’. On searching ‘champagnepapi’ first, I was able to gather only 230 entries, so, I went ahead with the next keyword. The next keyword was ‘drizzy’. I was able to gather 995 entries however most of the tweets were memes or football(soccer) related data. Finally, after using the keyword ‘drake’, I got entries with relevant data. All this data was collected in R Studio using the following code snippet:

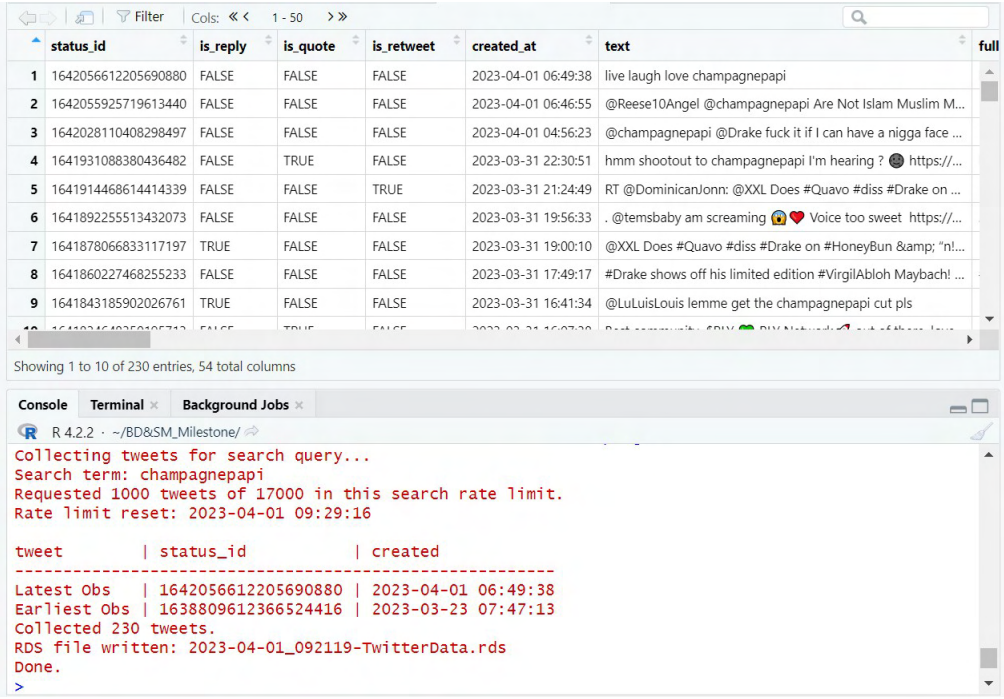
Figure 1

Code snippet for Data Retrieval

```
# Authenticate to Twitter and collect data

twitter_data <- Authenticate("twitter",
                             appName = my_app_name,
                             apiKey = my_api_key,
                             apiSecret = my_api_secret,
                             accessToken = my_access_token,
                             accessTokenSecret = my_access_token_secret) %>%
  Collect(searchTerm = "drake",
          searchType = "recent",
          numTweets = 1000,
          lang = "en",
          includeRetweets = TRUE,
          writeToFile = TRUE,
          verbose = TRUE) # use 'verbose' to show download progress
```

Figure 2

Data Retrieval when keyword was 'champagnepapi'



status_id	is_reply	is_quote	is_retweet	created_at	text	full
1	1642056612205690880	FALSE	FALSE	FALSE	2023-04-01 06:49:38	live laugh love champagnepapi
2	1642055925719613440	FALSE	FALSE	FALSE	2023-04-01 06:46:55	@Reese10Angel @champagnepapi Are Not Islam Muslim M...
3	1642028110408298497	FALSE	FALSE	FALSE	2023-04-01 04:56:23	@champagnepapi @Drake fuck it if I can have a nigga face ...
4	1641931088380436482	FALSE	TRUE	FALSE	2023-03-31 22:30:51	hmm shootout to champagnepapi I'm hearing ? https://...
5	1641914468614414339	FALSE	FALSE	TRUE	2023-03-31 21:24:49	RT @DominicanJonn: @XXL Does #Quavo #diss #Drake on ...
6	1641892255513432073	FALSE	FALSE	FALSE	2023-03-31 19:56:33	. @temsbaby am screaming Voice too sweet https://...
7	1641878066833117197	TRUE	FALSE	FALSE	2023-03-31 19:00:10	@XXL Does #Quavo #diss #Drake on #HoneyBun & "n!...
8	1641860227468255233	FALSE	FALSE	FALSE	2023-03-31 17:49:17	#Drake shows off his limited edition #VirgilAbloh Maybach! ...
9	1641843185902026761	TRUE	FALSE	FALSE	2023-03-31 16:41:34	@LuLuisLouis lemme get the champagnepapi cut pls

Showing 1 to 10 of 230 entries, 54 total columns

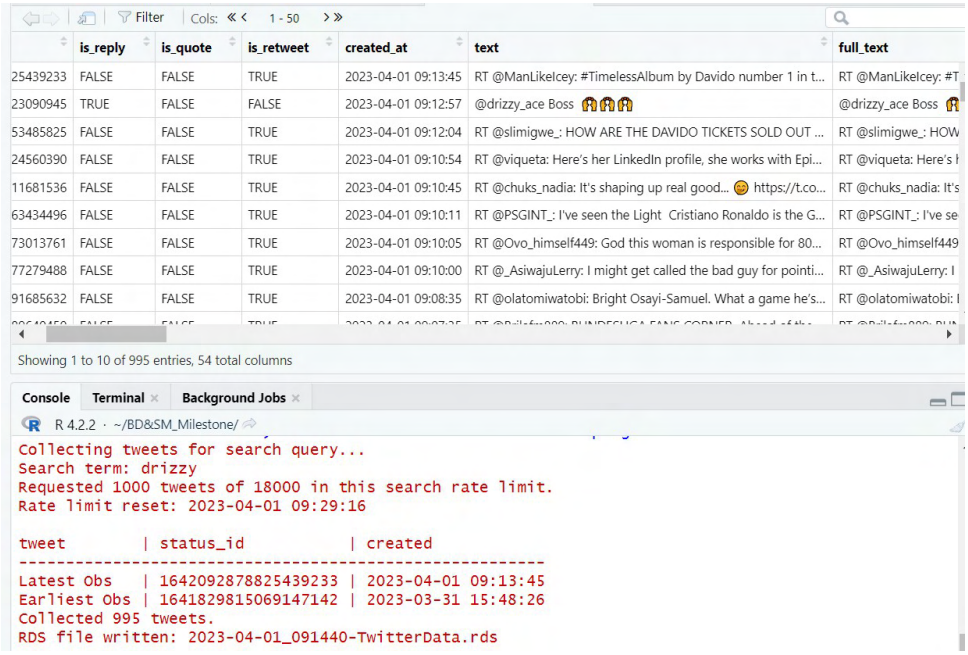
Console Terminal Background Jobs

R 4.2.2 · ~/BD&SM_Milestone/

```
Collecting tweets for search query...
Search term: champagnepapi
Requested 1000 tweets of 17000 in this search rate limit.
Rate limit reset: 2023-04-01 09:29:16

tweet        | status_id           | created
-------------------------------------------------------
Latest Obs   | 1642056612205690880 | 2023-04-01 06:49:38
Earliest Obs | 1638809612366524416 | 2023-03-23 07:47:13
Collected 230 tweets.
RDS file written: 2023-04-01_092119-TwitterData.rds
Done.
>
```

Figure 3

Data Retrieval when keyword was 'drizzy'



	is_reply	is_quote	is_retweet	created_at	text	full_text
25439233	FALSE	FALSE	TRUE	2023-04-01 09:13:45	RT @ManLikeIcey: #TimelessAlbum by Davido number 1 in t...	RT @ManLikeIcey: #T
23090945	TRUE	FALSE	FALSE	2023-04-01 09:12:57	@drizzy_ace Boss	@drizzy_ace Boss
53485825	FALSE	FALSE	TRUE	2023-04-01 09:12:04	RT @slimigwe_: HOW ARE THE DAVIDO TICKETS SOLD OUT ...	RT @slimigwe_: HOW
24560390	FALSE	FALSE	TRUE	2023-04-01 09:10:54	RT @viqueta: Here's her LinkedIn profile, she works with Epi...	RT @viqueta: Here's h
11681536	FALSE	FALSE	TRUE	2023-04-01 09:10:45	RT @chuks_nadia: It's shaping up real good... https://t.co...	RT @chuks_nadia: It's
63434496	FALSE	FALSE	TRUE	2023-04-01 09:10:11	RT @PSGINT_: I've seen the Light Cristiano Ronaldo is the G...	RT @PSGINT_: I've se
73013761	FALSE	FALSE	TRUE	2023-04-01 09:10:05	RT @Ovo_himself449: God this woman is responsible for 80...	RT @Ovo_himself449
77279488	FALSE	FALSE	TRUE	2023-04-01 09:10:00	RT @_AsiwajuLerry: I might get called the bad guy for pointi...	RT @_AsiwajuLerry: I
91685632	FALSE	FALSE	TRUE	2023-04-01 09:08:35	RT @olatomiwatobi: Bright Osayi-Samuel. What a game he's...	RT @olatomiwatobi: I

Showing 1 to 10 of 995 entries, 54 total columns

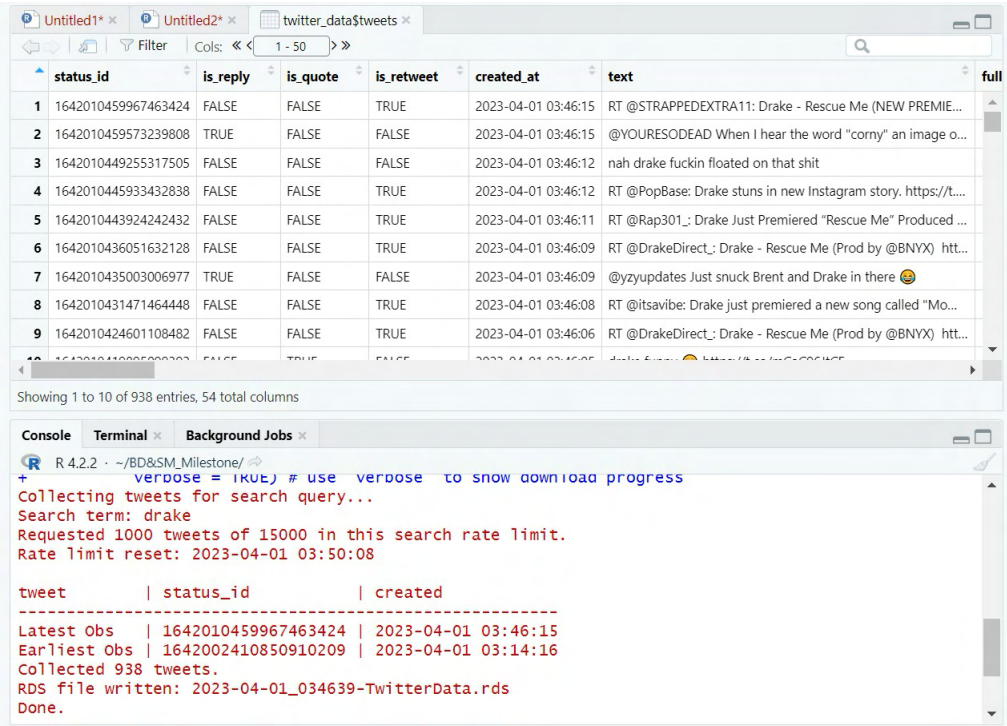
Console Terminal Background Jobs

R 4.2.2 · ~/BD&SM_Milestone/

```
Collecting tweets for search query...
Search term: drizzy
Requested 1000 tweets of 18000 in this search rate limit.
Rate limit reset: 2023-04-01 09:29:16

tweet        | status_id           | created
-------------------------------------------------------
Latest Obs   | 1642092878825439233 | 2023-04-01 09:13:45
Earliest Obs | 1641829815069147142 | 2023-03-31 15:48:26
Collected 995 tweets.
RDS file written: 2023-04-01_091440-TwitterData.rds
```

Figure 4

Data Retrieval when keyword was 'drake'



Untitled1* | Untitled2* | twitter_data$tweets

Filter | Cols: « < 1 - 50 > »

	status_id	is_reply	is_quote	is_retweet	created_at	text	full
1	1642010459967463424	FALSE	FALSE	TRUE	2023-04-01 03:46:15	RT @STRAPPEDEXTRA11: Drake - Rescue Me (NEW PREMIE...	
2	1642010459573239808	TRUE	FALSE	FALSE	2023-04-01 03:46:15	@YOURESODEAD When I hear the word "corny" an image o...	
3	1642010449255317505	FALSE	FALSE	FALSE	2023-04-01 03:46:12	nah drake fuckin floated on that shit	
4	1642010445933432838	FALSE	FALSE	TRUE	2023-04-01 03:46:12	RT @PopBase: Drake stuns in new Instagram story. https://t....	
5	1642010443924242432	FALSE	FALSE	TRUE	2023-04-01 03:46:11	RT @Rap301_: Drake Just Premiered "Rescue Me" Produced ...	
6	1642010436051632128	FALSE	FALSE	TRUE	2023-04-01 03:46:09	RT @DrakeDirect_: Drake - Rescue Me (Prod by @BNYX) htt...	
7	1642010435003006977	TRUE	FALSE	FALSE	2023-04-01 03:46:09	@yzyupdates Just snuck Brent and Drake in there 😂	
8	1642010431471464448	FALSE	FALSE	TRUE	2023-04-01 03:46:08	RT @itsavibe: Drake just premiered a new song called "Mo...	
9	1642010424601108482	FALSE	FALSE	TRUE	2023-04-01 03:46:06	RT @DrakeDirect_: Drake - Rescue Me (Prod by @BNYX) htt...	

Showing 1 to 10 of 938 entries, 54 total columns

Console | Terminal | Background Jobs

R 4.2.2 · ~/BD&SM_Milestone/

```
+          verbose = TRUE) # use "verbose" to show download progress
Collecting tweets for search query...
Search term: drake
Requested 1000 tweets of 15000 in this search rate limit.
Rate limit reset: 2023-04-01 03:50:08

tweet        | status_id           | created
-------------------------------------------------------
Latest Obs   | 1642010459967463424 | 2023-04-01 03:46:15
Earliest Obs | 1642002410850910209 | 2023-04-01 03:14:16
Collected 938 tweets.
RDS file written: 2023-04-01_034639-TwitterData.rds
Done.
```

Q. 1.3: List the top 5 most influential users for your artist/band. Find out what other interests/characteristics they have besides those related to your artist/band. Do these 5 have something in common? *(=> Lab 2.1)* [2.5 marks]

ANS.

The top 5 most influential users (usernames)are: Kurrco, Rap301_, STRAPPEDEXTRA11, DrakeDirect_, BNYX. Of these accounts, Kurrco, Rap301_ are hip-hop news/fan pages, DrakeDirect is a Drake fan page. BNYX is a music producer and all 5 of them have shared a post on a remix of a song made by BNYX.

Figure 5

Code snippet along with result for top 5 influential users

```
# Q 1.3: Finding top 5 most influential users
# Overwrite the 'name' attribute in your graph with the 'screen name' attribute
# to replace twitter IDs with more meaningful names,
# then run the Page Rank algorithm again

V(twitter_actor_graph)$name <- V(twitter_actor_graph)$screen_name

rank_twitter_actor <- sort(page_rank(twitter_actor_graph)$vector, decreasing = TRUE)
head(rank_twitter_actor, n = 5)
```

56:46 (Untitled) R Script

Console Terminal Background Jobs

R 4.2.2 · ~/BD&SM_Milestone/

```
> # Q 1.3
> # Overwrite the 'name' attribute in your graph with the 'screen name' attribute
> # to replace twitter IDs with more meaningful names,
> # then run the Page Rank algorithm again
>
> V(twitter_actor_graph)$name <- V(twitter_actor_graph)$screen_name
>
> rank_twitter_actor <- sort(page_rank(twitter_actor_graph)$vector, decreasing = TRUE)
> head(rank_twitter_actor, n = 5)
        Kurrco         Rap301_ STRAPPEDEXTRA11    DrakeDirect_            BNYX
    0.12117656      0.04417861      0.04176080      0.04070276      0.03634936
>
```

Figure 6

Kurrco (First user with the post tagging BNYX)

Figure 7

Rap301_ (second user with the post tagging BNYX)

Figure 8

STRAPPEDEXTRA11 (third user with the post tagging BNYX)

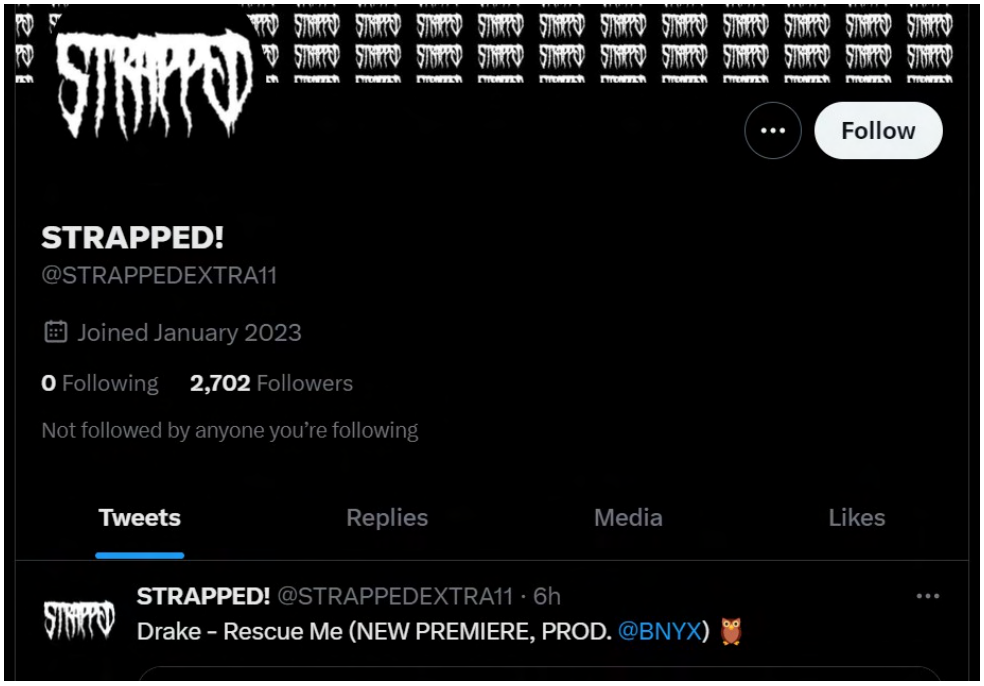


Figure 9

DrakeDirect_ (fourth user with the post tagging BNYX)

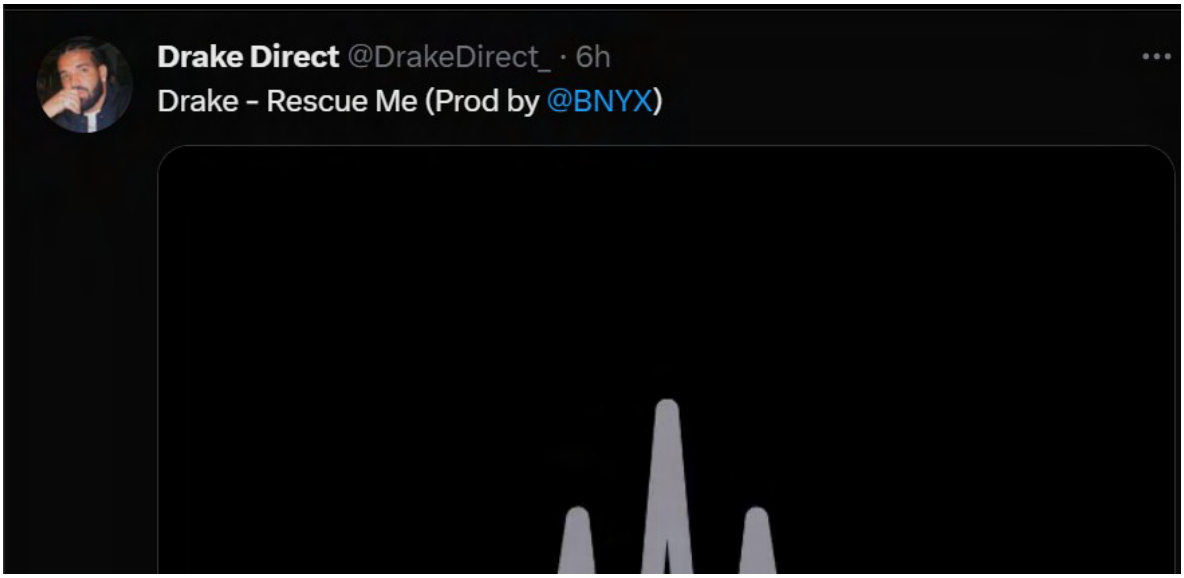


Figure 10

BNYX (Fifth User)

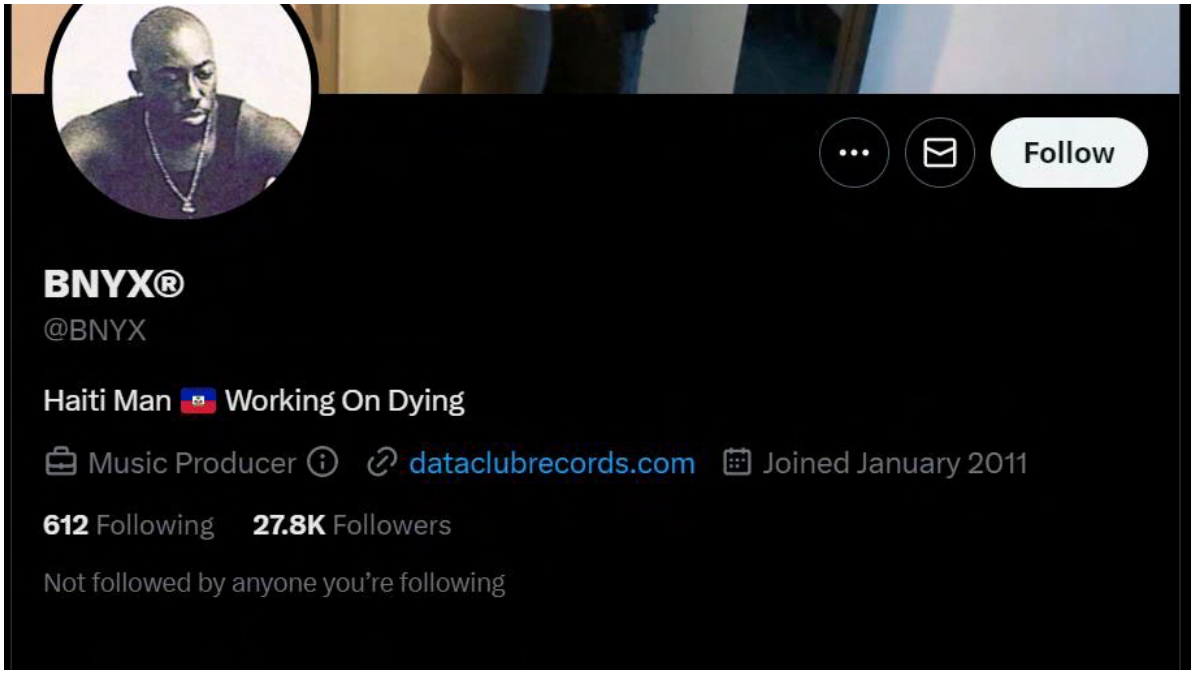
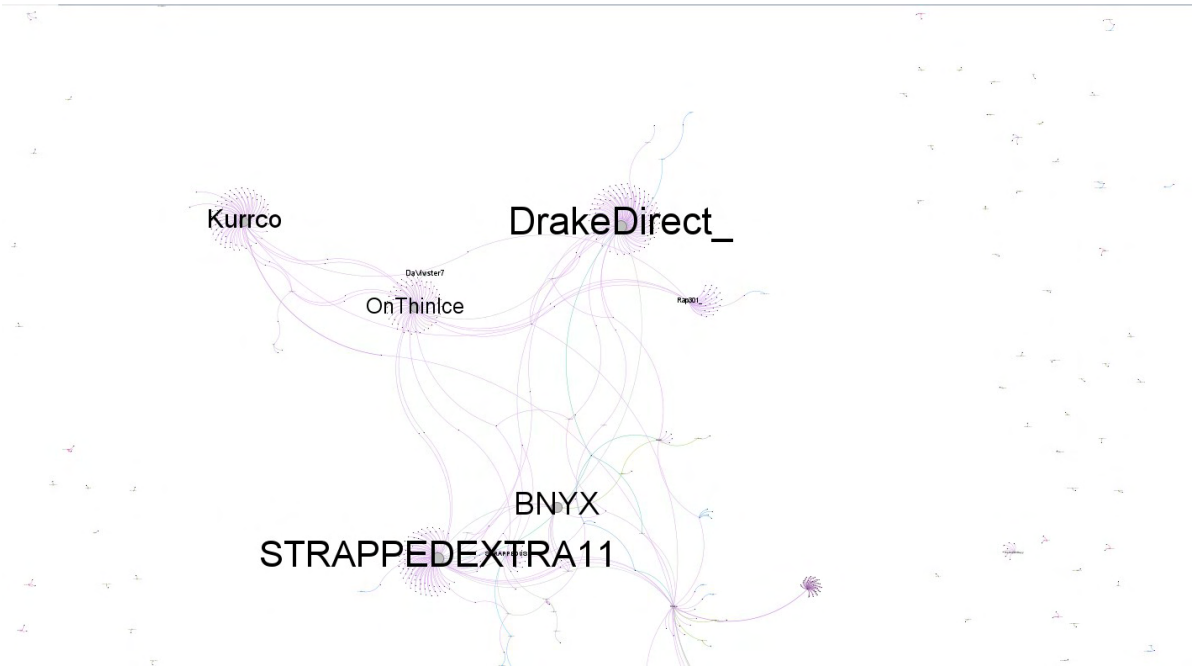


Figure 11

Top 5 influential users (and others) (Twitter Actor Graph)



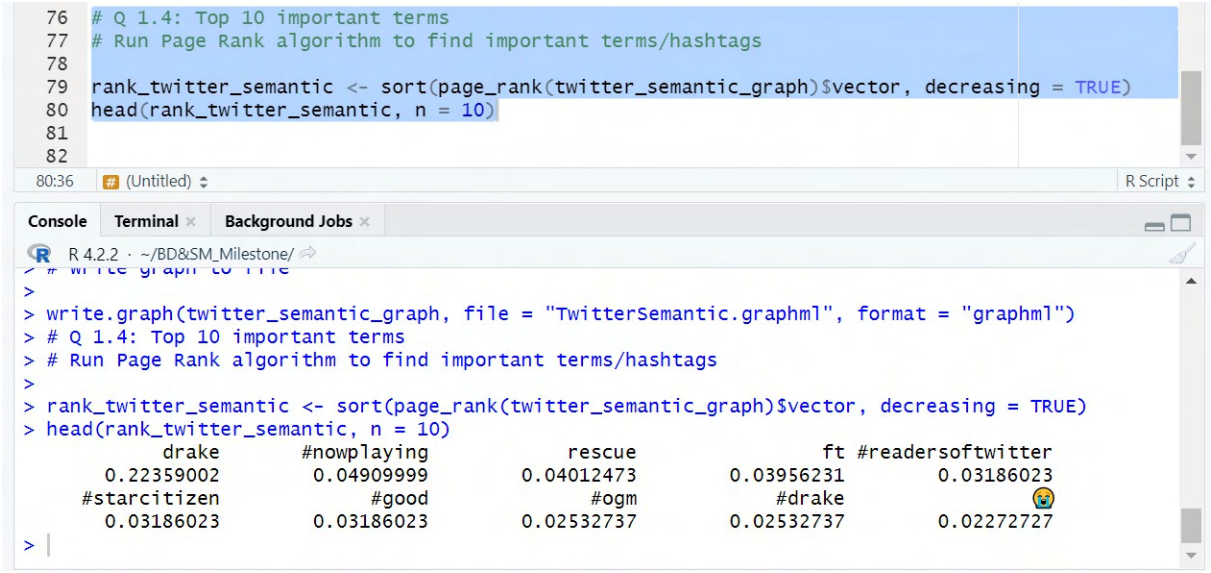
Q. 1.4: List the top 10 most important terms that appear together with your keyword(s) related to your artist/band. Explain the results. *(=> Lab 2.1)* [1.5 marks]

ANS.

The 10 most important terms that appear together with drake:

Figure 12

Top 10 important terms



Keywords drake, #nowplaying, rescue, ft & #drake are relevant keywords as they relate to the remix of a song named ‘Rescue’.

Figure 13

Graph for the top 10 topics

Q. 1.5: Calculate how many unique user accounts there are in your dataset. Explain the code you have used for the calculation. What do the results tell you? [2.5 marks]

ANS.

There are 1100 unique users in my dataset. I got this data from exporting the data into a csv file in Gephi (TwitterActor file). Then I used ‘count’ function to count how many times a screen name was used to find unique users.

Figure 14

Snippet of code used to find unique user accounts

```
# Q.1.5
# Finding the unique users of keyword 'drake' dataset

library('dplyr')
library("data.table")

datainquestion <- read.csv("C:/Users/omkar/OneDrive/Documents/BD&SM_Milestone/DrakeRetrievedData.csv")

results <- datainquestion %>%
  group_by(v_screen_name) %>%
  summarize(count = n_distinct(v_screen_name))

#view results
results

length(unique(results$count))

#view graph of results
plot.default(results)
```

Figure 15

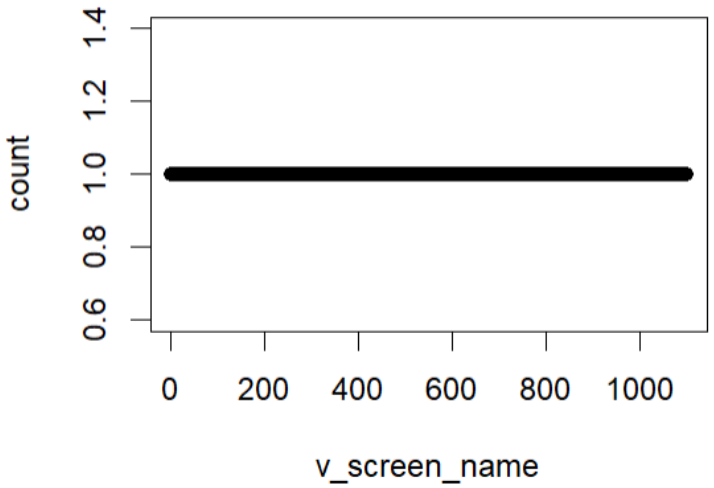
Table of result containing unique users

Filter

	v_screen_name	count
1089	youtubemusic	1
1090	ysela__	1
1091	yslnorman	1
1092	yungintrovertz	1
1093	yzyslds	1
1094	yzyupdates	1
1095	zachluvsroblox	1
1096	zackatk27	1
1097	zaederr	1
1098	zahtheDonDada	1
1099	zen0m	1
1100	zeta_mecha	1

Figure 16

Graph of unique users


count
0.6
0.8
1.0
1.2
1.4
0
200
400
600
800
1000
v_screen_name

Data Selection & Exploration (continued)

2.1) Use the Spotify API to extract data about your artist/band. For example:

○ How many years have they been active?

○ How many albums & songs have they published?

○ With whom have they often collaborated?

○ What are the prevalent features of their songs (e.g., valence)?

How does the Spotify data compare to the information you collected from other sources in Step 1.1 (Milestone 1)? *(=> Lab 2.2)* [1.8 marks]

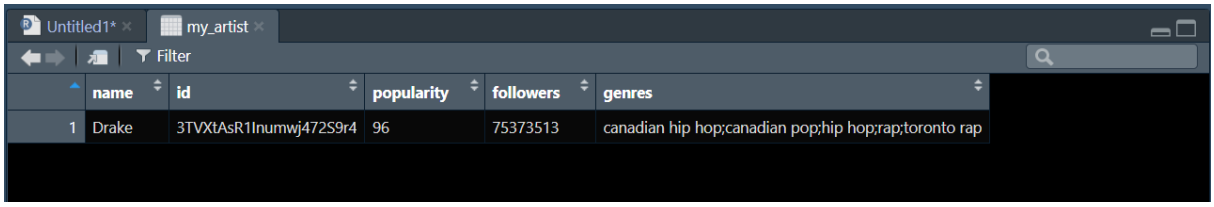
ANS.

Artist Information was found by using the following query:

my_artist <- getArtist("3TVXtAsR1Inumwj472S9r4", token = keys)

View(my_artist)

The result was:



	name	id	popularity	followers	genres
1	Drake	3TVXtAsR1Inumwj472S9r4	96	75373513	canadian hip hop;canadian pop;hip hop;rap;toronto rap

Album information was found on using:

albums <- getAlbums("3TVXtAsR1Inumwj472S9r4", token = keys)

View(albums)

The result was:

22	5mz0mJxb80gqJlcRf9LGHJ	Nothing Was The Same (Deluxe)	album
23	3s8Lsh6OjP2uWK6UYaTzEv	Nothing Was The Same (Deluxe)	album
24	1XsIirSxfAhhxRdn4Li9t	Nothing Was The Same	album
25	3PAv4PLPv9V68fstzIqhyb	Nothing Was The Same	album
26	6X1x82kppWZmDzIXXK3y3q	Take Care (Deluxe)	album
27	4epK17BZv559EXBACsQXQG	Take Care (Deluxe)	album
28	6jlrjFR9mJV3jd1IPSplXU	Thank Me Later	album
29	5EReyE1t1d2tzKwwATAqTo	Thank Me Later	album
30	1LShhEEKRT5MNPcO7jtYHh	So Far Gone	album

Showing 11 to 30 of 30 entries, 4 total columns

According to this, Drake has published 30 albums. However, the data on the snippet shows the same album with different album IDs because some albums may be normal, some explicit albums or same may even be special editions with a few extra tracks.

To get distinct album names, following command was

used: unique(albums$name)

The result was:

```
> unique(albums$name)
 [1] "Her Loss"                                "Honestly, Nevermind"
 [3] "Certified Lover Boy"                     "Dark Lane Demo Tapes"
 [5] "Care Package"                            "So Far Gone"
 [7] "Scorpion"                                "More Life"
 [9] "Views"                                   "What A Time To Be Alive"
[11] "If You're Reading This It's Too Late"    "Nothing Was The Same (Deluxe)"
[13] "Nothing Was The Same"                    "Take Care (Deluxe)"
[15] "Thank Me Later"
>
```

The result was 15 unique albums out of which one album is deluxe version of another (Nothing was The Same) which makes number of albums to 14 which is same number mentioned in the Milestone 1.

By searching for Audio features, the prominent audio features, number of songs released, and some other data was generated. The following code was used to get that:

#Audio features

audio_features <- get_artist_audio_features("Drake")

View(audio_features)

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

View(audio_features)

unique(audio_features$track_nam

e) The number of songs found were

247:

```
[235] "Karaoke"
[236] "The Resistance"
[237] "Over"
[238] "Show Me A Good Time"
[239] "Up All Night"
[240] "Fancy"
[241] "Shut It Down"
[242] "Unforgettable"
[243] "Light Up"
[244] "Miss Me"
[245] "Cece's Interlude"
[246] "Find Your Love"
[247] "Thank Me Now"
>
```

The audio features were:

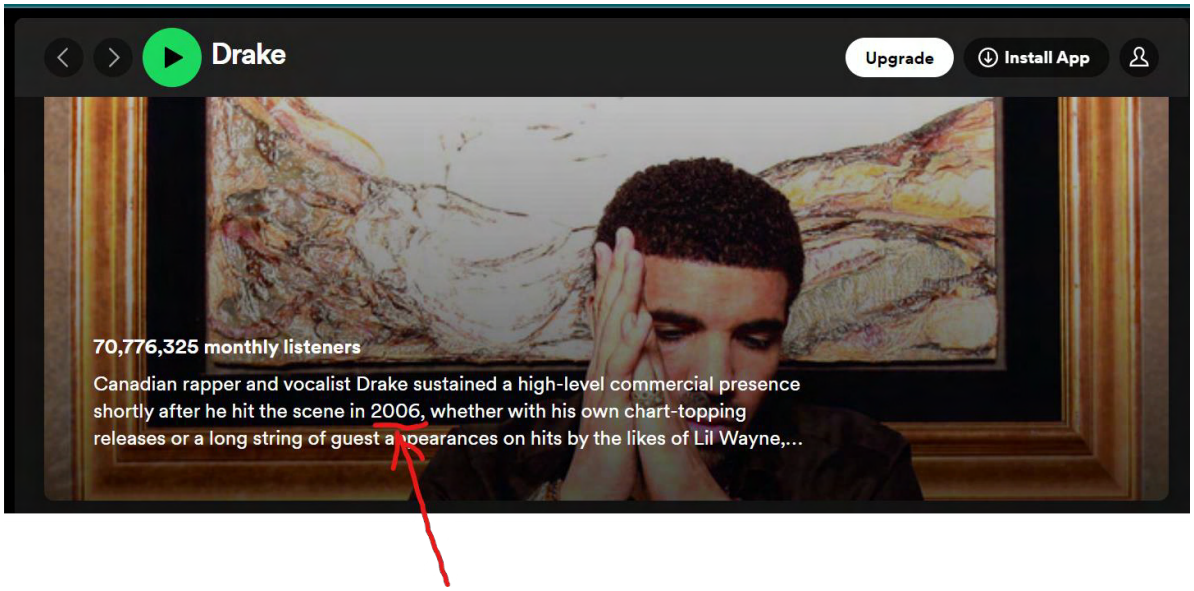
lbum_images	album_release_date	album_release_year	album_release_date_precision	danceability	energy	key	loudness	mo
variables	2022-11-04	2022	day	0.561	0.5200	11	-9.342	
variables	2022-11-04	2022	day	0.908	0.5460	8	-10.491	
variables	2022-11-04	2022	day	0.841	0.3580	9	-8.368	
variables	2022-11-04	2022	day	0.849	0.4330	5	-8.434	
variables	2022-11-04	2022	day	0.934	0.6140	5	-7.384	
variables	2022-11-04	2022	day	0.773	0.7010	7	-6.386	
variables	2022-11-04	2022	day	0.501	0.4030	6	-11.106	
variables	2022-11-04	2022	day	0.746	0.5170	10	-7.582	
variables	2022-11-04	2022	day	0.734	0.6050	1	-8.476	
variables	2022-11-04	2022	day	0.749	0.6300	4	-6.652	
variables	2022-11-04	2022	day	0.642	0.5330	2	-7.450	
variables	2022-11-04	2022	day	0.466	0.7030	4	-5.774	
variables	2022-11-04	2022	day	0.799	0.5750	6	-6.257	
variables	2022-11-04	2022	day	0.704	0.5040	1	-6.749	
variables	2022-11-04	2022	day	0.475	0.6920	1	-8.327	
variables	2022-11-04	2022	day	0.385	0.3400	7	-11.754	
variables	2022-06-17	2022	day	0.193	0.0218	0	-31.160	
variables	2022-06-17	2022	day	0.718	0.7580	10	-8.290	
variables	2022-06-17	2022	day	0.765	0.6270	0	-4.607	

Showing 1 to 19 of 247 entries, 39 total columns

When filtered by album release date, the oldest song released in an album was found to be in 2010.

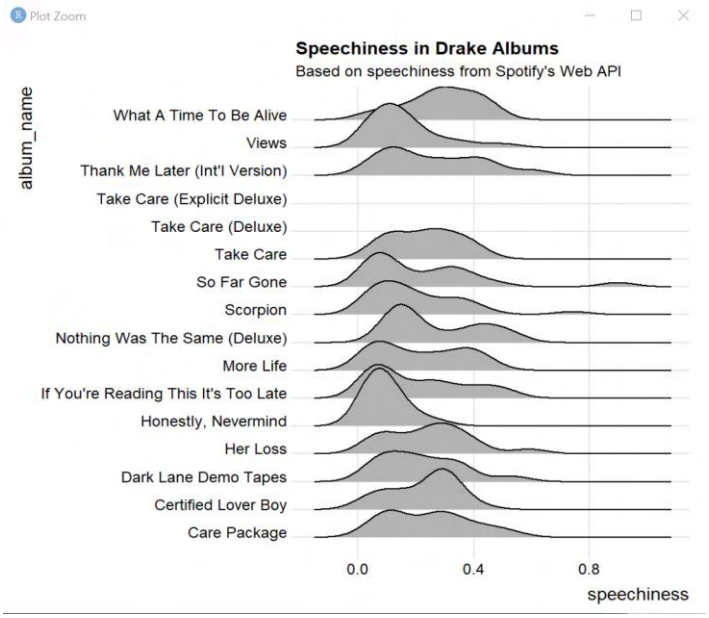
	artist_name	artist_id	album_id	album_type	album_images	album_release_date
588	Drake	3TVXtAsR1Inumwj472S9r4	6agmeioaDOBupymziJihgB	album	*2 variables*	2010-01-01

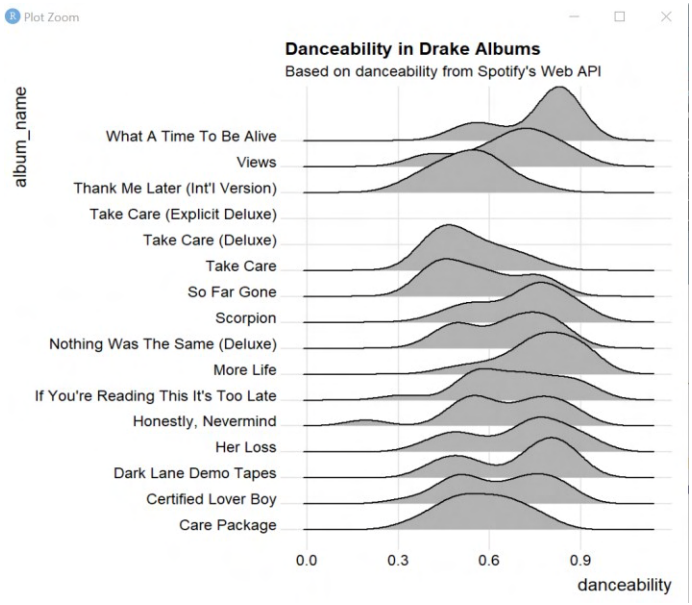
However, on his Spotify homepage, its mentioned that he began his career in 2006. This information is matching with the information retrieved in milestone 1.

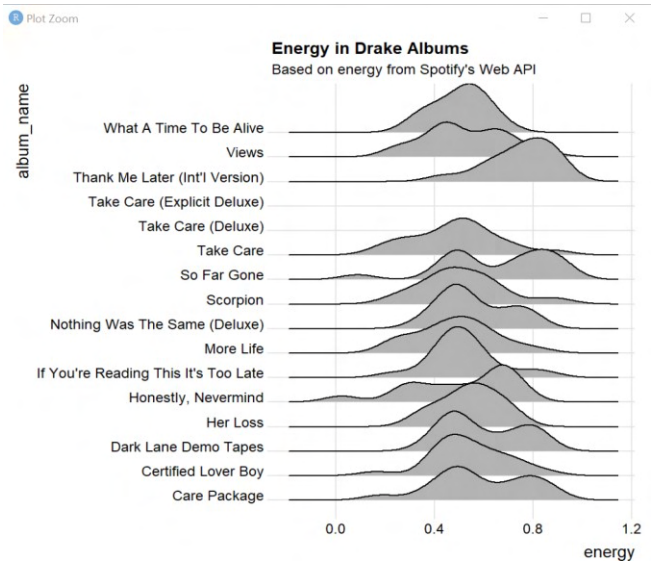


The prevalent features of his songs were mapped using the following code: ggplot(audio_features, aes(x = speechiness, y = album_name)) + geom_density_ridges() +
theme_ridges() +
ggtitle("Speechiness in Drake Albums",
subtitle = "Based on speechiness from Spotify's Web API") ggplot(audio_features, aes(x = danceability, y = album_name)) + geom_density_ridges() +
theme_ridges() +
ggtitle("Danceability in Drake Albums",
subtitle = "Based on danceability from Spotify's Web API")

ggplot(audio_features, aes(x = energy, y = album_name)) +
geom_density_ridges() +
theme_ridges() +
ggtitle("Energy in Drake Albums",
subtitle = "Based on energy from Spotify's Web API") The results were:





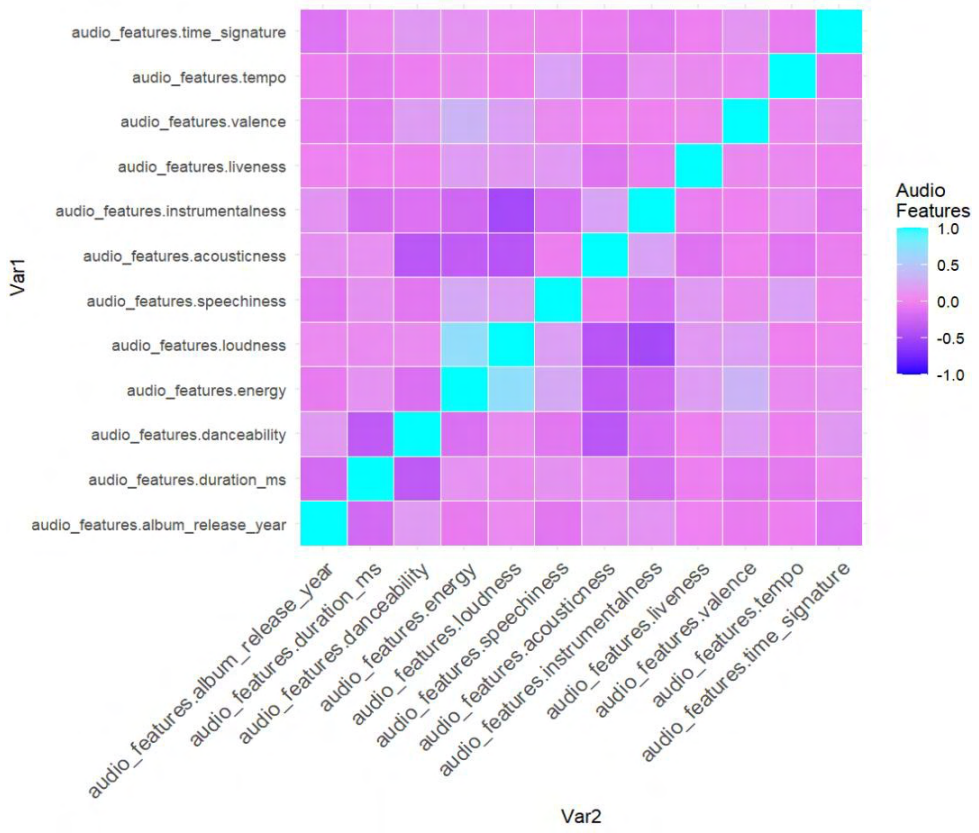


A correlation graph was also created on the audio features extracted from the query above. A data frame was created on the columns taken in from the audio features result. The following queries were used to get a corelation heatmap:

```
#Corr Plot of All audio features

af_df <- data.frame(audio_features$album_release_year, audio_features$duration_ms, audio_features$danceability, audio_features$energy, audio_features$loudness, audio_features$speechiness, audio_features$acousticness, audio_features$instrumentalness, audio_features$liveness, audio_features$valence, audio_features$tempo, audio_features$time_signature)

library(corrplot)

cormat =

cor(af_df)
```

```
corrplot((cor(af_df)))
library(reshape2)
melted_data <- melt(cor(af_df))
reorder_cormat <-
function(cormat){
 # Use correlation between variables as
 distance dd <- as.dist((1-cormat)/2)
 hc <- hclust(dd)
 cormat <-cormat[hc$order, hc$order]
}
# Reorder the correlation
matrix cormat <-
reorder_cormat(cormat)
upper_tri <-
get_upper_tri(cormat) # Melt
the correlation matrix
melted_cormat <- melt(upper_tri, na.rm =
TRUE) # Create a ggheatmap
ggheatmap <- ggplot(melted_data, aes(Var2, Var1, fill = value))+
 geom_tile(color = "white")+
 scale_fill_gradient2(low = "blue", high = "cyan", mid =
                "violet", midpoint = 0, limit = c(-1,1), space =
                "Lab",
                name="Audio\nFeatures") +
 theme_minimal()+ # minimal theme
 theme(axis.text.x = element_text(angle = 45, vjust = 1,
                          size = 12, hjust = 1))+
 coord_fixed()
# Print the heatmap
print(ggheatmap)
```

The resultant heatmap was:



Related artists were found using query:

related_bm <- getRelated("Drake", token = keys) View(related_bm)

Milestone2Script.R* | related_bm | top_5L | top_5V

Filter

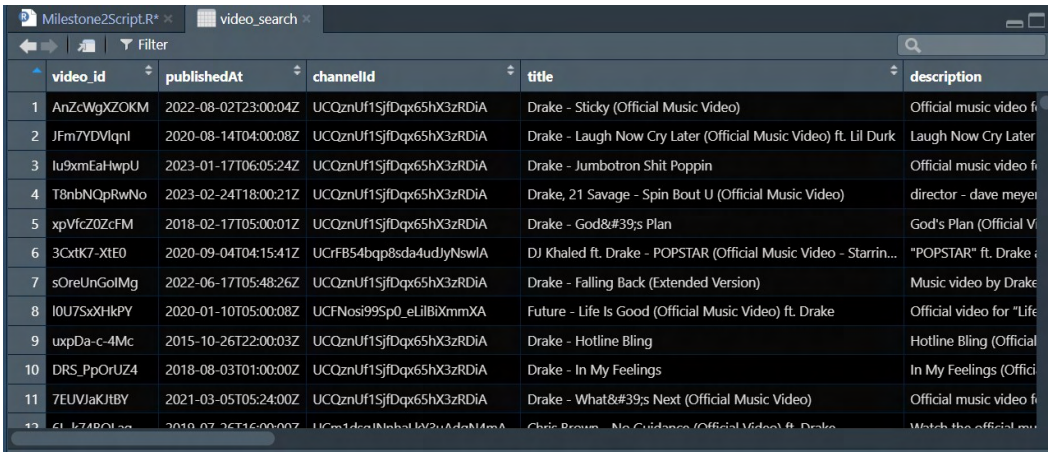
	name	id	popularity	type	followers
1	Big Sean	0c173mlxpT3dSFRgMO8XPh	77	artist	10824411
2	J. Cole	6l3HvQ5sa6mXTsMTB19rO5	87	artist	20164337
3	DJ Khaled	0QHgL1lAIqAw0HtD7YldmP	77	artist	9739150
4	Meek Mill	20sxb77xiYeusSH8cVdatc	75	artist	7224077
5	Future	1RyvyyTE3xzB2ZywiAwp0i	90	artist	13903566
6	PARTYNEXTDOOR	2HPaUgqeutzr3jx5a9WyDV	78	artist	5180010
7	Tory Lanez	2jku7tDXc6XoB6MO2hFuqg	80	artist	5508501
8	Young Thug	50co4Is1HCEo8bhOyUWKpn	84	artist	8327152
9	Rick Ross	1sBkRIssrMs1AbVkOJbc7a	77	artist	6802928
10	21 Savage	1URnnhqYAYcrqrcwql10ft	91	artist	13706173
11	2 Chainz	17lzZA2AlOHwCwFALHttmp	75	artist	7830951
12	Bryson Tiller	2EMAnMvWE2eb56ToJVfCWs	79	artist	6964686

2.2) Retrieve data relevant to your artist/band from YouTube. Which videos have the highest number of views and likes? Do you see a correlation between views and likes? (Your dataset may contain hundreds of videos, so it's OK if you choose only a subset of those to get their statistics, in order to avoid hitting the rate-limit. However, you should get statistics for at least 5 videos.) (=> Lab 3.2) [1.8 marks]

ANS.

The YouTube data was extracted using this query:

video_search <- yt_search("Drake Official Music Video")



	video_id	publishedAt	channelId	title	description
1	AnZcWgXZOKM	2022-08-02T23:00:04Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - Sticky (Official Music Video)	Official music video f
2	JFm7YDVlqnI	2020-08-14T04:00:08Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - Laugh Now Cry Later (Official Music Video) ft. Lil Durk	Laugh Now Cry Later
3	Iu9xmEaHwpU	2023-01-17T06:05:24Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - Jumbotron Shit Poppin	Official music video f
4	T8nbNQpRwNo	2023-02-24T18:00:21Z	UCQznUf1SjfDqx65hX3zRDiA	Drake, 21 Savage - Spin Bout U (Official Music Video)	director - dave meyer
5	xpVfcZ0ZcFM	2018-02-17T05:00:01Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - God's Plan	God's Plan (Official V
6	3CxtK7-XtE0	2020-09-04T04:15:41Z	UCrFB54bqp8sda4udJyNswlA	DJ Khaled ft. Drake - POPSTAR (Official Music Video - Starrin...	"POPSTAR" ft. Drake
7	sOreUnGolMg	2022-06-17T05:48:26Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - Falling Back (Extended Version)	Music video by Drake
8	l0U7SxXHkPY	2020-01-10T05:00:08Z	UCFNosi99Sp0_eLilBiXmmXA	Future - Life Is Good (Official Music Video) ft. Drake	Official video for "Life
9	uxpDa-c-4Mc	2015-10-26T22:00:03Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - Hotline Bling	Hotline Bling (Official
10	DRS_PpOrUZ4	2018-08-03T01:00:00Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - In My Feelings	In My Feelings (Offici
11	7EUVJaKJtBY	2021-03-05T05:24:00Z	UCQznUf1SjfDqx65hX3zRDiA	Drake - What's Next (Official Music Video)	Official music video f

To find channel stats, the following query was used:

chan <- get_all_channel_video_stats(channel_id =

"UCQznUf1SjfDqx65hX3zRDiA") To find the top 5 most viewed and liked

videos, following queries were used:

library(sqldf)

top_5V <- sqldf("SELECT id,title,viewCount,likeCount,url FROM chan

ORDER BY CAST(viewCount AS numeric) DESC

LIMIT 5", row.names =

TRUE) View(top_5V)

The result was:

	id	title	viewCount	likeCount	url
1	uxpDa-c-4Mc	Drake - Hotline Bling	1929849767	10599642	https://www.youtube.com/watch?v=uxpDa-c-4Mc
2	xpVfcZ0ZcFM	Drake - God's Plan	1490544692	15451332	https://www.youtube.com/watch?v=xpVfcZ0ZcFM
3	RubBzkZzpUA	Drake - Started From The Bottom	507818811	2683564	https://www.youtube.com/watch?v=RubBzkZzpUA
4	JFm7YDVlqnI	Drake - Laugh Now Cry Later (Official Music Video) ft. Lil Durk	441100325	4049473	https://www.youtube.com/watch?v=JFm7YDVlqnI
5	-zzP29emgpg	Drake - Take Care ft. Rihanna	430464677	2068819	https://www.youtube.com/watch?v=-zzP29emgpg

Similarly, for top 5 most liked:

top_5L <- sqldf("SELECT id,title,viewCount,likeCount,url FROM

chan ORDER BY CAST(likeCount AS numeric) DESC

```
    LIMIT 5", row.names =
TRUE) View(top_5L)
```

The result was:

Milestone2Script.R* | top_5L | top_5V

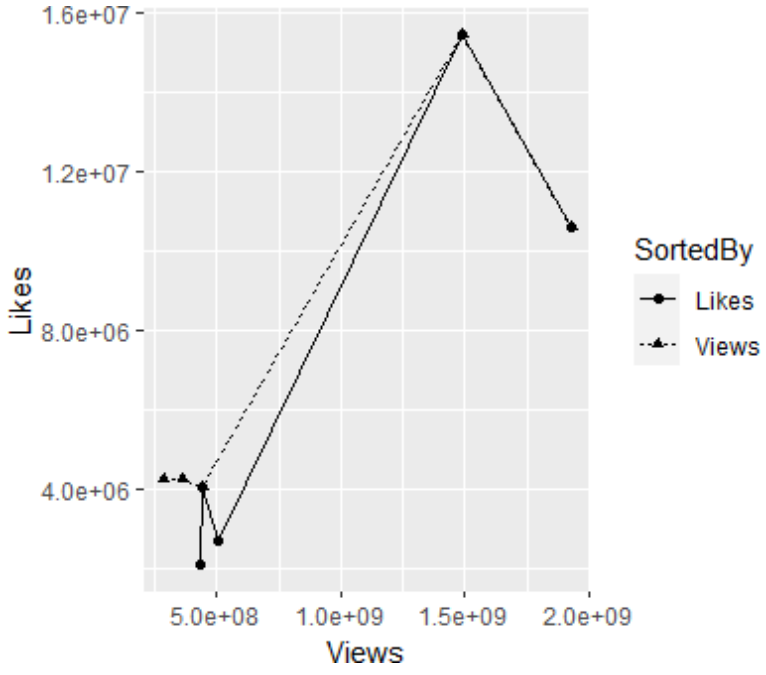
Filter

	id	title	viewCount	likeCount	url
1	xpVfcZ0ZcFM	Drake - God's Plan	1490544692	15451332	https://www.youtube.com/watch?v=xpVfcZ0ZcFM
2	uxpDa-c-4Mc	Drake - Hotline Bling	1929849767	10599642	https://www.youtube.com/watch?v=uxpDa-c-4Mc
3	DRS_PpOrUZ4	Drake - In My Feelings	287923761	4247348	https://www.youtube.com/watch?v=DRS_PpOrUZ4
4	xWggTb45brM	Drake - Toosie Slide (Official Music Video)	360484852	4234106	https://www.youtube.com/watch?v=xWggTb45brM
5	JFm7YDVIqnI	Drake - Laugh Now Cry Later (Official Music Video) ft. Lil Durk	441100325	4049473	https://www.youtube.com/watch?v=JFm7YDVIqnI

Despite having higher Views, the Highest number of likes for videos was different.

Then I plotted a graph to show relationships between videos with top views vs videos with top likes using the following code:

```
df <- data.frame(SortedBy=rep(c("Likes", "Views"), each=5),
                 Views=c(1929552454, 1490332372, 507785195, 440935088, 430373338, 1490332372, 1929552454, 287923761, 360484852, 440935088),
                 Likes=c(10597918, 15449108, 2683277, 4048613, 2068505, 15449108, 10597918, 4247348, 4234106, 4048613))
ggplot(df, aes(x=Views, y=Likes, group=SortedBy)) +
  geom_line(aes(linetype=SortedBy))+
  geom_point(aes(shape=SortedBy))
```



Text Pre-Processing

2.3) Perform text pre-processing and create a Term-Document Matrix for your Twitter data. What are the 10 terms occurring with the highest frequency? How are they different to your answer for Step 1.4 (Milestone 1)? (=> Lab 2.2) [1.8 marks]

ANS. The last line of the code used for getting frequency of top 10 most occurring terms is:

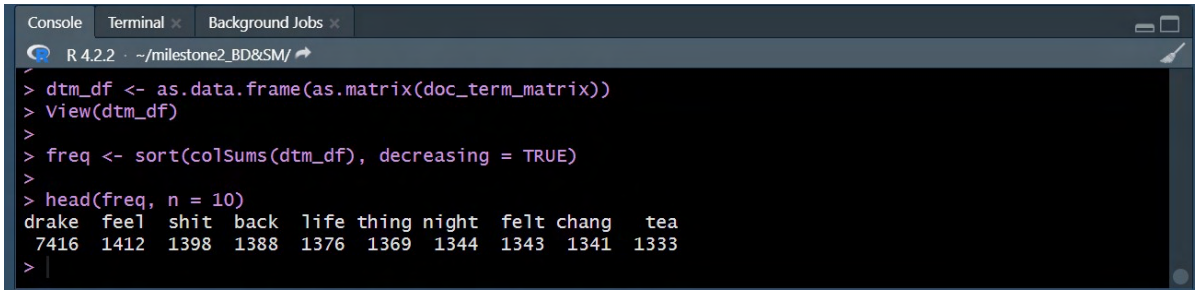
dtm_df <- as.data.frame(as.matrix(doc_term_matrix))

View(dtm_df)

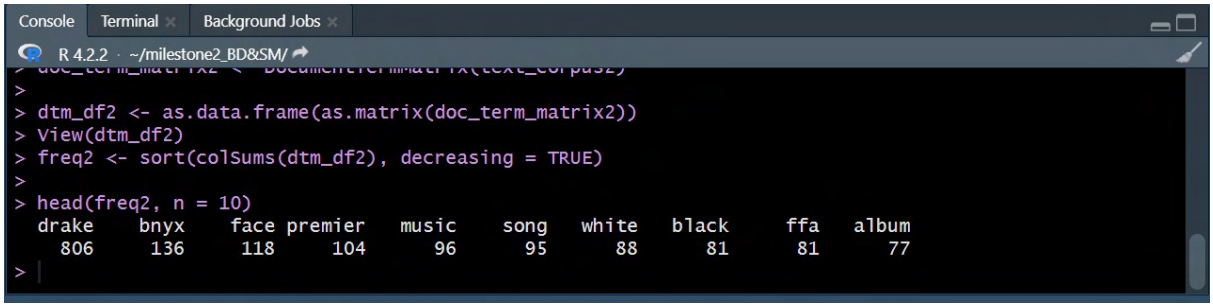
freq <- sort(colSums(dtm_df), decreasing = TRUE)

head(freq, n = 10)

The output was:

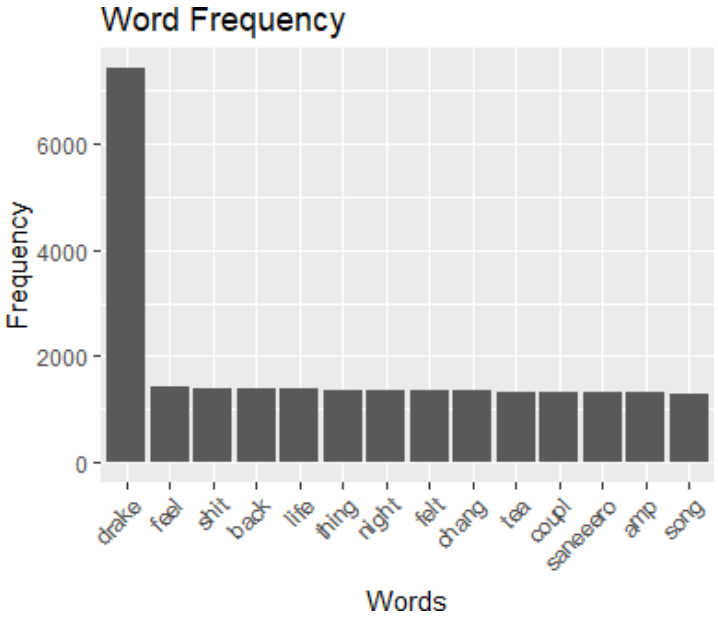


The top 10 terms in Milestone 1 were:

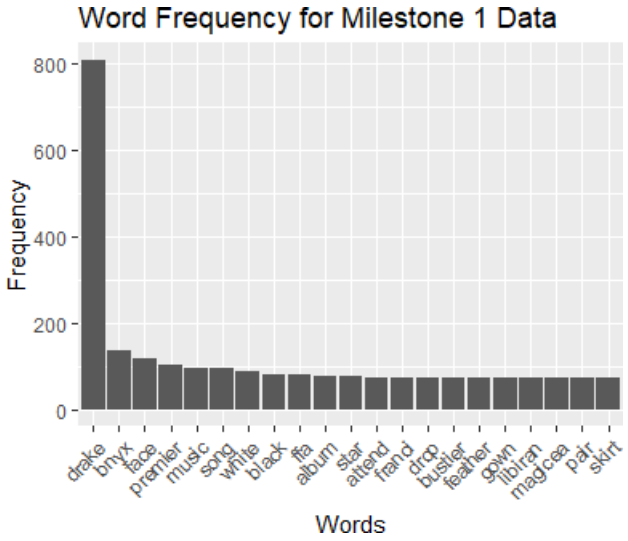


The terms in milestone 1 and milestone 2 are very different. This might be due to the gap in between taking both the datasets and a remix song was launched during the data collection of milestone 1 so that data is about the remix.

The plot for word frequency when using Data for milestone 2:



The plot for word frequency when using Data for milestone 1:



Social Network Analysis

2.4) Perform centrality analysis by detecting degree centrality, betweenness centrality, and closeness centrality. Explain how relevant the results are to your artist/band. What is the actual degree, betweenness, and closeness centrality scores for your artist/band node in the network? Compare these scores to the scores for related artists. *(=> Lab 3.1)* [3.6 marks]

ANS. The Centrality analysis for this milestone is performed on Twitter data of Drake, DJ Khaled, and Justin Bieber. The degree centrality refers to which node is the most connected in the network. The node with the greatest number of edges (incoming or outgoing) has the highest degree centrality. The degree centrality on data for milestone 2 is as follows:

Mode = “in” stands for number of incoming edges. Mode = “out” stands for number of outgoing edges. Mode = “total” stands for number of any edges either in or out. The numbers below each term refer to number of edges for that node.

```
> sort(degree(twomode_subgraph, mode = "in"), decreasing = TRUE)[1:20]
       @drake     @chartdata @taylorswift13  @justinbieber   @mariahcarey       @madonna
          376            164            130             85             75             73
  @thebeatles       @rihanna @eltonofficial  @steviewonder  @janetjackson     @theweeknd
           73             73             73             73             73             59
        @sza    @elpesopluma        @karolg     @sanbenito       @ferxxo4  @morganwallen
           45             43             43             41             41             41
  @lanadelrey    @spotifydata
           41             39
> sort(degree(twomode_subgraph, mode = "out"), decreasing = TRUE)[1:20]
  @nonniequeen5    @josephstitman       @lada_dala @hernandez_jency       @mvgm_1224
            111              76              66              52              48
   @nastymilana @sinatradasniper         @tr5pico  @legendarytingz        @pddddd33
             47              44              39              37              35
@nanette86951434       @mvp_drake    @michuemenalo    @calebkaminga    @jereme_drake
             34              34              33              32              31
       @ka182736       @aiverdoun       @cam___c13 @an0therlanastan      @tayebmaraj
             29              26              26              25              25
> sort(degree(twomode_subgraph, mode = "total"), decreasing = TRUE)[1:20]
         @drake       @chartdata   @taylorswift13    @nonniequeen5    @justinbieber
            376              164              130              113               85
  @josephstitman     @mariahcarey         @madonna      @thebeatles         @rihanna
             76               75               73               73               73
  @eltonofficial    @steviewonder    @janetjackson       @lada_dala       @theweeknd
             73               73               73               66               59
@hernandez_jency @sinatradasniper     @nastymilana       @mvgm_1224            @sza
             52               52               49               48               45
```

Now for Closeness Centrality, Closeness Centrality refers to the shortest path between two nodes in a network. The closeness centrality score is usually between 0 and 1. The closer it is to 1, the closer it is to other nodes in the network.

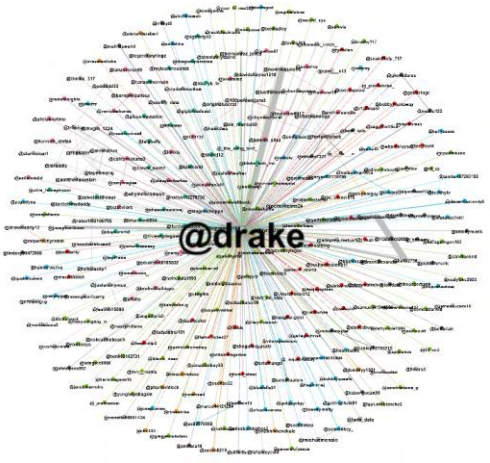
```
> sort(closeness(twomode_subgraph, mode = "in"), decreasing = TRUE)[1:20]
@douglas50266581  @decembertwnty8        @shottasv       @exxon8219 @californialana3
               1               1               1               1               1
  @vitorspeaknow        @pddddd33       @capp52719     @alisha12287         @tr5pico
               1               1               1               1               1
    @exileoutdid @bornpinkjulissa      @screeeaamm    @calebkaminga          @ssbxlv
               1               1               1               1               1
   @thedreadgaze      @nfthuntguy        @modic123          @tdo667    @eziko1571008
               1               1               1               1               1
> sort(closeness(twomode_subgraph, mode = "out"), decreasing = TRUE)[1:20]
 @moreabtnothing    @iamdjshotgunn  @ziggywalksinla      @gothcringe     @stetsonestes
               1               1               1               1               1
@theglammorelife  @utkarshmisra12 @villainmagazine     @virgo96_123      @daachecker
               1               1               1               1               1
      @eusarasou        @a1a2mami      @bolidarity       @tee_fount @officialjjazmin
               1               1               1               1               1
    @draco500005       @trezortre          @stilez     @turki_atawi     @shawn_don13
               1               1               1               1               1
> sort(closeness(twomode_subgraph, mode = "total"), decreasing = TRUE)[1:20]
         @drake     @lilmike_317  @irreverentlabs  @alexanderebert     @sirchtheweb
    0.0003114295     0.0002632965     0.0002441406     0.0002431907     0.0002429543
 @cokeatthebeach         @jeeefzz   @justanonymus_  @revengeonfoamy    @leslihh15800
    0.0002416626     0.0002416626     0.0002414293     0.0002410800     0.0002410800
  @ladyastral101          @priahx @bruhwtfisthisyo    @remy_daystar @eduardo49105032
    0.0002410800     0.0002409639     0.0002409639     0.0002409639     0.0002409639
      @omankooo_   @petterhelms24 @davidshayne1210      @irinaoma16         @hawtdwa
    0.0002409639     0.0002407898     0.0002399808     0.0002398082     0.0002397507
```

Now for Betweenness Centrality, it measures how frequently a node comes in the shortest path between two other nodes. The nodes with higher betweenness centrality scores are considered more influential in connecting other nodes.

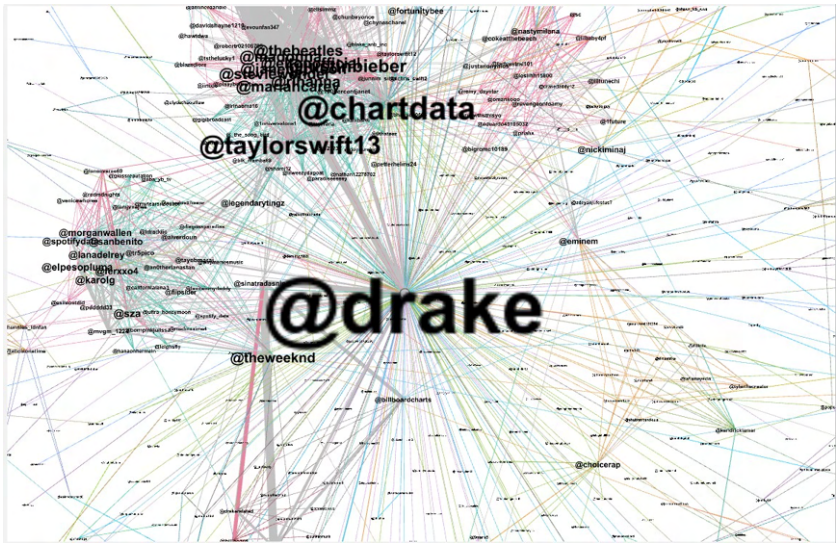
```
> sort(betweenness(twomode_subgraph, directed = FALSE), decreasing = TRUE)[1:20]
          @drake     @lilmike_317           #music       @souljaboy       @smsboomer               #ai
       465113.63        157335.83         80846.85         60960.00         60003.00         59313.28
      @elonmusk    @jereme_drake        @youtube      #nowplaying         @jeeefzz    @michuemenalo
        59101.00         58043.88         49414.40         48992.69         39048.34         33873.00
 @irreverentlabs    @talibandztbg       #theweeknd       @chartdata    @endwokeness @alexanderebert
        32648.88         31561.48         30113.68         30067.31         27626.22         25034.81
     @kimzymine @eurojournaleng
        24406.00         24348.49
>
```

Visualization in Gephi,

This is the Degree Centrality Plot of the Drake Twitter data set. This was obtained by running Network Diameter and Average Degree in the Statistics Tab of Gephi. Then further filtering the appearance by Degree Centrality.



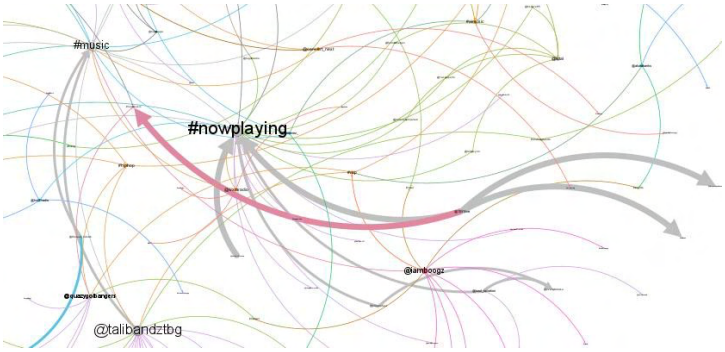
On zooming in the above graph, we get,



The terms visible here are also mentioned in the R studio results

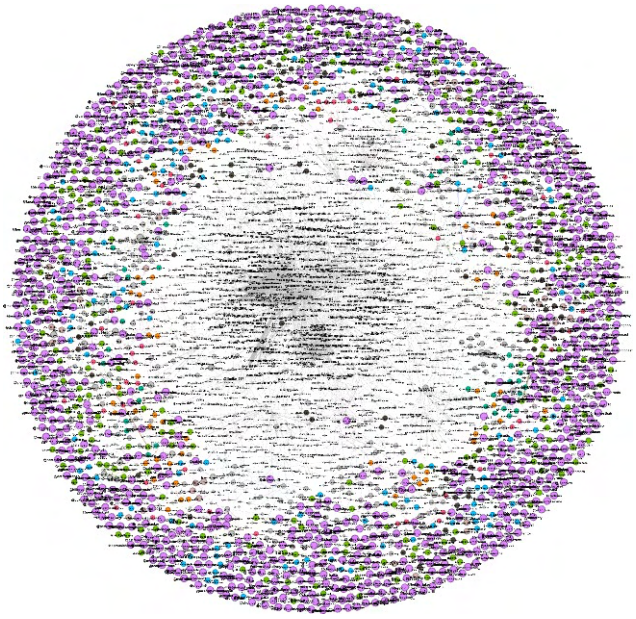
above. On further filtering with giant component and degree rank,

we get,

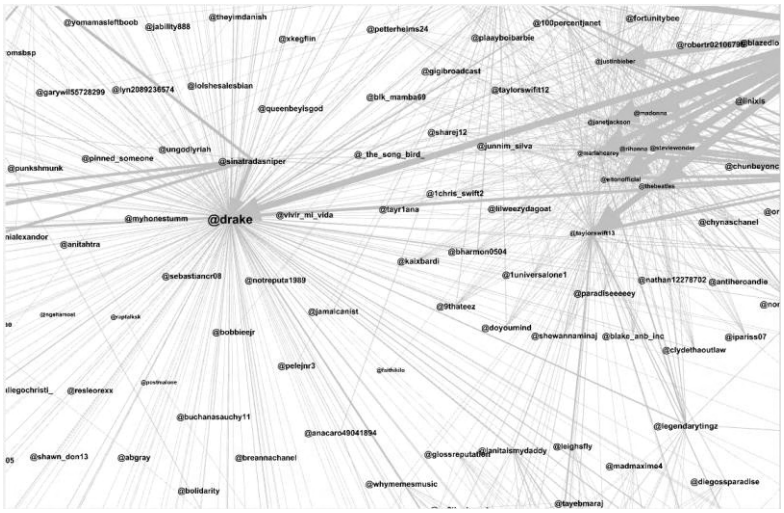


This snippet shows a small part of whole degree centrality when '#nowplayying' was one of the keywords for Milestone 1 when a remix of a Drake song was released.

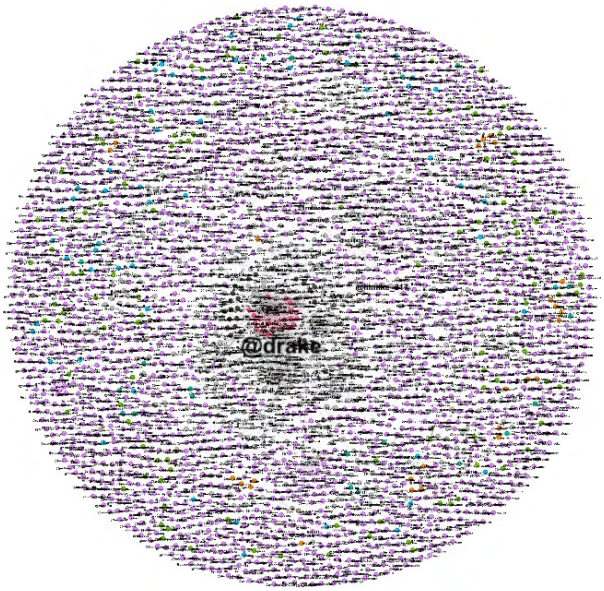
Now, for Closeness Centrality, we run Network Diameter and filtering the appearance by Closeness Centrality.



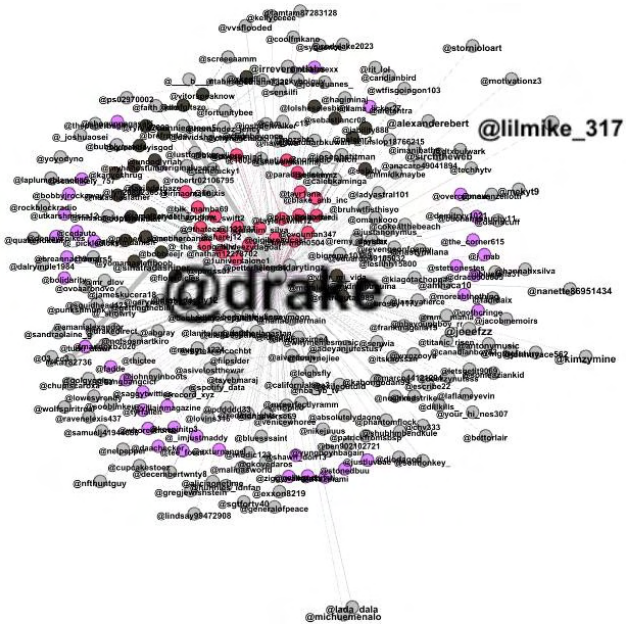
When filtered by Shortest path between 2 nodes, we get,



Now, for Betweenness Centrality, we run Network Diameter and filtering the appearance by Betweenness Centrality.



On filtering by Ego Network with node = n1375, we get,



Similarly, when performed the Centrality Analysis for DJ Khaled, we get,

Degree Centrality as,

```
> sort(degree(twomode_subgraph2, mode = "in"), decreasing = TRUE)[1:20]
       @youtube        @djkhaled        @rickross      @liltunechi        #djkhaled
             28                8                4                4                4
       @lildurk              @sc      @johnlegend         #jordan5           #retro
              3                3                3                2                2
     #wethebest      @lilbaby4pf        @wnrv1081          @rtfkt       @fightmate
              2                2                2                2                1
    @powermommy2     @adamcarolla @adamcarollashow           @drake   #mohammedsiraj
              1                1                1                1                1
> sort(degree(twomode_subgraph2, mode = "out"), decreasing = TRUE)[1:20]
     @hbradio_lv      @sellerhub4       @wnrv1081_    @lilwaynehq_2     @blognatives
              15                8                8                4                4
        @djdelz    @c_millertime @blusoonerredram  @theyogiestbeer @wristaficionado
              3                3                3                3                3
    @shmaeli999   @catherine4785 @dangelowiiliams   @finessekraken   @disruptappeal
              3                2                1                1                1
   @realdillonrw      @hhazelboyy       @ryoryo719     @rashiedlane       @ghaly1199
              1                1                1                1                1
> sort(degree(twomode_subgraph2, mode = "total"), decreasing = TRUE)[1:20]
       @youtube      @hbradio_lv      @sellerhub4       @wnrv1081_        @djkhaled
             28               15                8                8                8
   @lilwaynehq_2     @blognatives        @rickross      @liltunechi        #djkhaled
              4                4                4                4                4
        @djdelz    @c_millertime @blusoonerredram  @theyogiestbeer @wristaficionado
              3                3                3                3                3
    @shmaeli999          @lildurk              @sc      @johnlegend   @catherine4785
              3                3                3                3                2
```

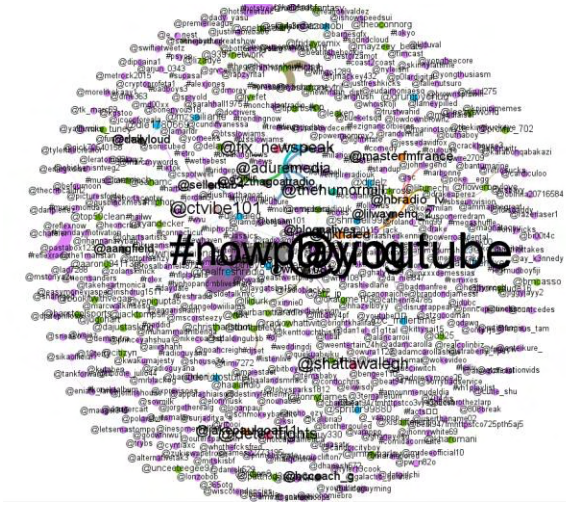
Closeness Centrality as,

```
> sort(closeness(twomode_subgraph2, mode = "in"), decreasing = TRUE)[1:20]
      @fightmate     @powermommy2     @adamcarolla @adamcarollashow           @drake
       1.0000000        1.0000000        1.0000000        1.0000000        1.0000000
  #mohammedsiraj           @gwaap           #rolex         #daydate            #boss
       1.0000000        1.0000000        1.0000000        1.0000000        1.0000000
       #shmaeli         #jordan5           #retro       #wethebest      @lilbaby4pf
       1.0000000        0.5000000        0.5000000        0.5000000        0.5000000
       @wnrv1081          @rtfkt         @lildurk              @sc      @johnlegend
       0.5000000        0.5000000        0.3333333        0.3333333        0.3333333
> sort(closeness(twomode_subgraph2, mode = "out"), decreasing = TRUE)[1:20]
@dangelowiiliams   @finessekraken   @disruptappeal    @realdillonrw      @hhazelboyy
               1                1                1                1                1
      @ryoryo719     @rashiedlane       @ghaly1199 @jinxxxdemessias      @badmanfree
               1                1                1                1                1
    @mscorsteezy   @alamokingofnc @brandondjamessf    @roland_cmg99       @0115_jack
               1                1                1                1                1
     @kojjomilan      @canonaiche   @sammydumsha23    @knighttrack1 @powerroaddental
               1                1                1                1                1
> sort(closeness(twomode_subgraph2, mode = "total"), decreasing = TRUE)[1:20]
       @youtube     @blognatives        @djkhaled         #djkhaled          @djdelz
     0.006250000      0.006097561      0.005319149      0.004854369      0.004716981
   @c_millertime      @shmaeli999   @finessekraken   @disruptappeal    @realdillonrw
     0.004716981      0.004716981      0.004629630      0.004629630      0.004629630
     @hhazelboyy       @ryoryo719     @rashiedlane       @ghaly1199 @jinxxxdemessias
     0.004629630      0.004629630      0.004629630      0.004629630      0.004629630
     @badmanfree   @alamokingofnc @brandondjamessf    @roland_cmg99       @0115_jack
     0.004629630      0.004629630      0.004629630      0.004629630      0.004629630
>
```

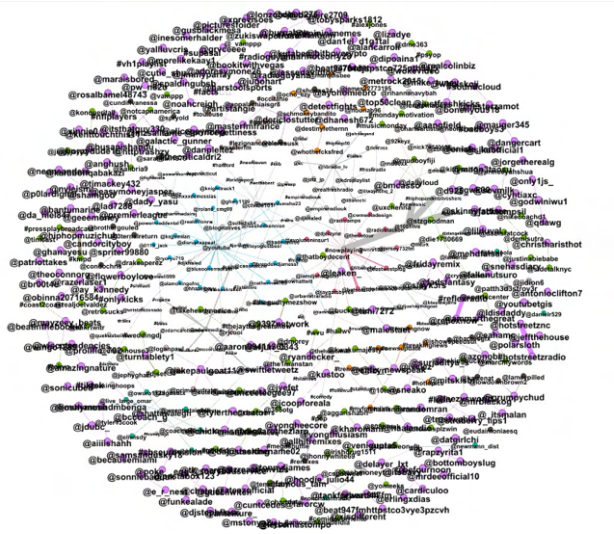
Betweenness Centrality as,

```
> sort(betweenness(twomode_subgraph2, directed = FALSE), decreasing = TRUE)[1:20]
       @youtube     @blognatives        @djkhaled         #djkhaled        @wnrv1081_
           1236              967              741              362              217
    @sellerhub4  @theyogiestbeer    @lilwaynehq_2          @djdelz    @c_millertime
            165              164              159              111              111
    @hbradio_lv @wristaficionado      @shmaeli999          @lildurk           @rtfkt
            111              111              111               56               56
      @rickross      @liltunechi @blusoonerredram @dangelowiiliams   @finessekraken
              2                2                0                0                0
>
```

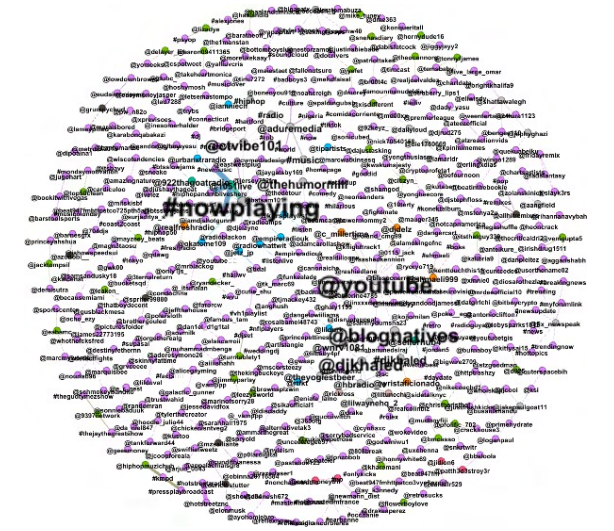
Similarly, the Visualization obtained was as follows, Degree Centrality as,



Closeness Centrality as,



Betweenness Centrality as,



Similarly, when performed the Centrality Analysis for Justin Bieber, we get, Degree Centrality as,

```
> sort(degree(twomode_subgraph3, mode = "in"), decreasing = TRUE)[1:20]
   #justinbieber      #selenagomez     #iheartawards     #haileybieber        #beliebers
            1013               319               225               184               171
    #taylorswift      #bestfanarmy    #kimkardashian        #kanyewest      #harrystyles
             160               144               142               138               136
      #videogame           #gamer          #gaming        #retrogame       #gamergirl
             135               135               135               135               134
@officialjbvotes             #bts       #blackpink #iheartawards2023    #neversaynever
             100                91                88                84                82
> sort(degree(twomode_subgraph3, mode = "out"), decreasing = TRUE)[1:20]
     @wevidgame       @mrpawmiaw      @nsk_online      @felicitrae          @dem469
           1079               790               371               200               200
@iheartawards23 @2023iheartradio @djdang3rousrajd @_bb_radio_music @manifestgurutwt
            185               184               182               114                93
  @notreality91   @adriantecharp    @dradrianwong @fanclub_biebers         @inbella
             83                77                76                68                62
@iheartradiomus3          @tpeide @obsessedduckman @handmadegd_etsy      @offmynovak
             61                59                50                50                44
> sort(degree(twomode_subgraph3, mode = "total"), decreasing = TRUE)[1:20]
     @wevidgame    #justinbieber       @mrpawmiaw      @nsk_online     #selenagomez
           1079             1013              790              371              319
  #iheartawards      @felicitrae          @dem469 @iheartawards23 @2023iheartradio
            225              200              200              185              184
  #haileybieber @djdang3rousrajd       #beliebers     #taylorswift     #bestfanarmy
            184              182              171              160              144
 #kimkardashian       #kanyewest     #harrystyles       #videogame           #gamer
            142              138              136              135              135
>
```

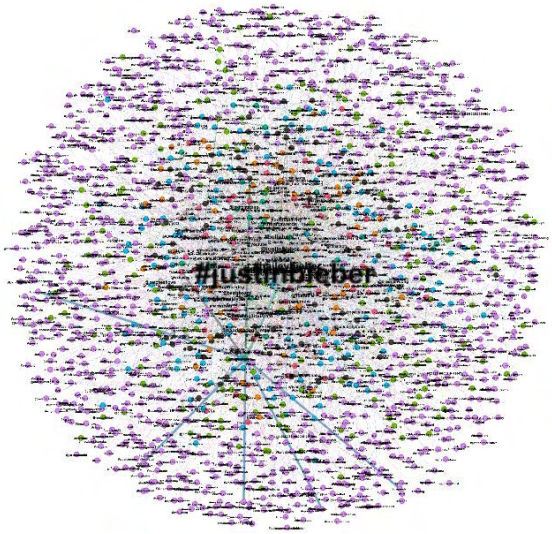
Closeness Centrality as,

```
> sort(closeness(twomode_subgraph3, mode = "in"), decreasing = TRUE)[1:20]
     @richlux713 @worldmusicaward   #inspirational       #guccifw23      @karlawelch
               1                1                1                1                1
        #nftart     #ukcelebrity        @drpepple        #playlist       #kimwoojin
               1                1                1                1                1
   @4everbrandy          @target    @rhinorecords          #brandy           #starz
               1                1                1                1                1
   @thepoptingz             #as      #tiktokdown    #tds2inlondon       @drewhouse
               1                1                1                1                1
> sort(closeness(twomode_subgraph3, mode = "out"), decreasing = TRUE)[1:20]
     @myminsmins       @polly8128   @oyosportsnews         @awonndy   @jagranenglish
               1                1                1                1                1
       @abplive       @4mjalways   @madhyamam_eng          @zoomtv    @editorbilkul
               1                1                1                1                1
   @bilkulonline     @ianslife_in       @goast_k1d   @channelrradio     @rektbidding
               1                1                1                1                1
   @biebtanfiles @juliett59778255     @lorenneesae  @sandeep_adnani            @much
               1                1                1                1                1
> sort(closeness(twomode_subgraph3, mode = "total"), decreasing = TRUE)[1:20]
   #justinbieber     @2muchbiebss   @gingertraceyb    @samsmith8971       @dunyazade
    0.0001287498     0.0001117568     0.0001111729     0.0001106439     0.0001102050
   @actressshana    @umar24671939 @kikhkampouridoy  @youtubeconten1    @levi_mccurdy
    0.0001098660     0.0001097213     0.0001096732     0.0001093016     0.0001089918
    @theshahab4  @christinesview @zukerbergtroper @oldscho72761891     @selenahails
    0.0001089325     0.0001089206     0.0001088969     0.0001088969     0.0001086838
 @morganfeetlove    @trendzone247    @thekellieray @londonbeautlife     @shiekababee
    0.0001079447     0.0001078283     0.0001077006     0.0001075731     0.0001075500
>
```

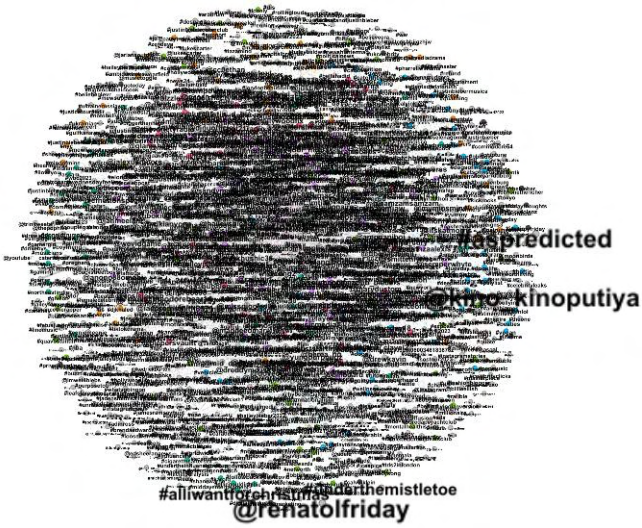
Betweenness Centrality as,

```
> sort(betweenness(twomode_subgraph3, directed = FALSE), decreasing = TRUE)[1:20]
   #justinbieber    #neversaynever          @youtube     @allaboutbndz     @actressshana
      1216896.80         274318.58         146626.23         140152.36          90485.97
    #selenagomez     @samsmith8971       #beliebers     #haileybieber @kikhkampouridoy
         85114.40          84204.05          71204.15          60851.86          58581.88
          #bieber          @inbella   #haileybaldwin              #dog     @notreality91
         52192.03          51202.21          50337.88          47910.37          47466.00
    @2muchbiebss     @justinbieber       @theshahab4       @mrpawmiaw      @timeblogxyz
         45638.80          43785.69          42633.89          40579.36          36987.09
>
```

Similarly, the Visualization obtained was as follows, Degree Centrality as,



Closeness Centrality as,



Betweenness centrality as,

2.5) Perform community analysis with the Girvan-Newman (edge betweenness) and Louvain methods. Explain how relevant the results are to your artist/band. Perform the community analysis also for related artists. Is their community structure similar? (=> Lab 3.2) [3.6 marks]

ANS. Louvain Algorithm is used to identify groups of nodes with strong internal connections and weak connections to nodes in other communities.

Girvan-Newman is also a community detection method that iteratively removes edges with high betweenness centrality to uncover the underlying community structure of a complex network. Once this is run, we can get to see hierarchy of the network when it was a giant community first and eventually becomes smaller communities.

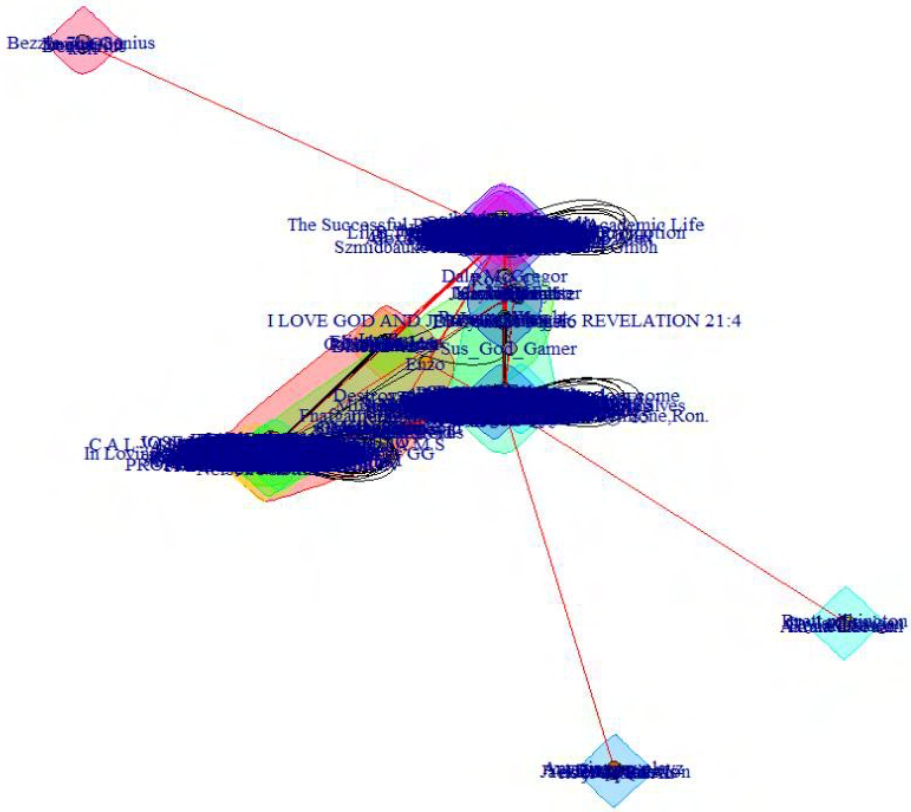
For this task, YouTube data for Drake, DJ Khaled and 21 Savage was used. *It was observed that all 3 artists had a similar community structure with 3 big communities and multiple small communities.*

Louvain Algorithm for Drake:

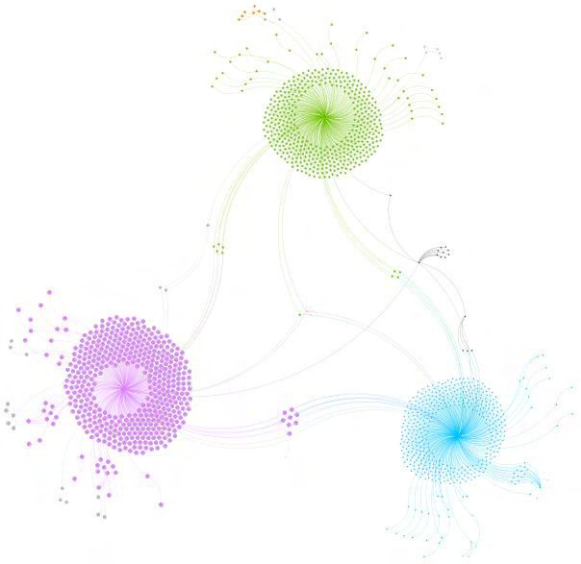
The community sizes obtained were, these sizes are the number of nodes in each community. From the result below, there are 3 large communities with 450+ nodes and 20 small communities.

```
> sizes(louvain_yt_actor)
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
457   5   2   5   3   2   1  15   2   2 466   5   2   7   3   3 465   2   3   3   2   3   5
>
```

The plot comes as:



When performing Louvain in Gephi, we can see the 3 large communities and other small communities.

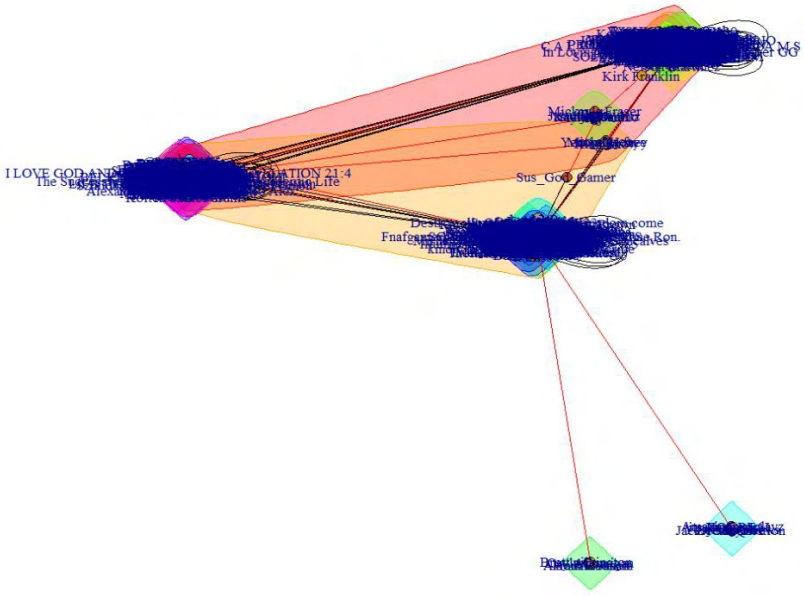


Girvan-Newman Algorithm for Drake:

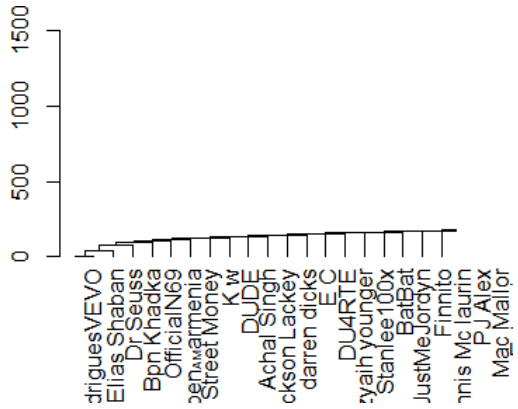
The community sizes obtained were, these sizes are the number of nodes in each community. From the result below, there are 3 large communities with 400+ nodes and 43 small communities.

```
> sizes(eb_yt_actor)
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
446   5   2   2   5 447   3   2   2   2   2   2   2  10   2   2   5   2   2   2   2   2   7   2
 25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46
  2   2   2   2   2   3   2   2 454   2   2   2   2   3   3   2   3   2   2   2   2   5
>
```

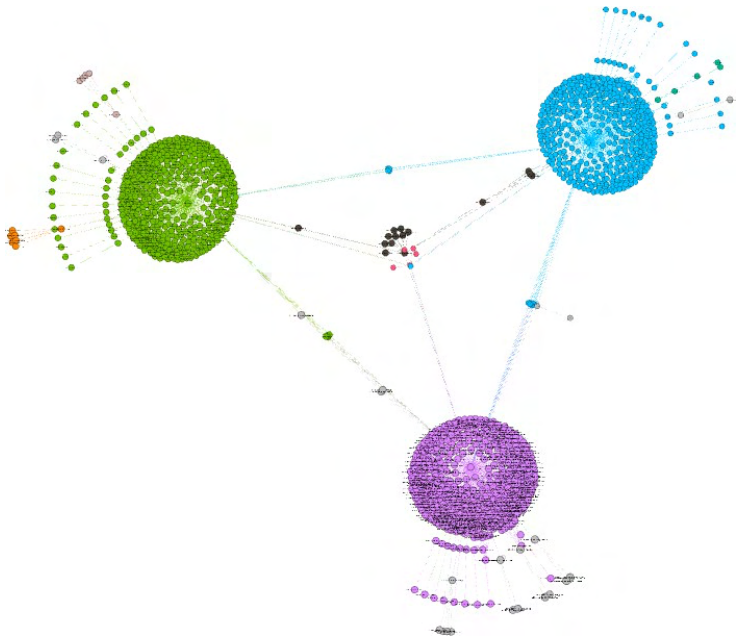
The plot comes as:



The visualization for edge-betweenness hierarchy:



When performing Girvan-Newman in Gephi, we can see the 3 large communities and other small communities.

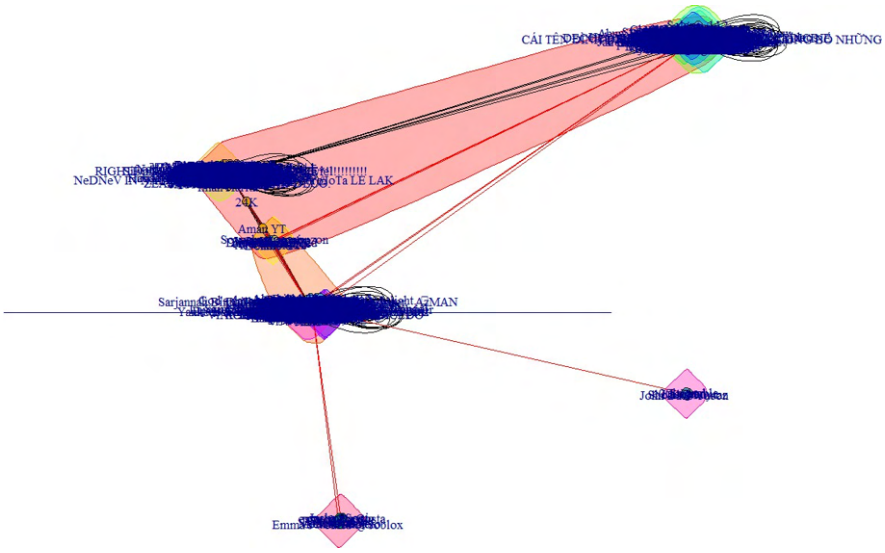


Similarly, when performed the Algorithms for DJ Khaled, we get,

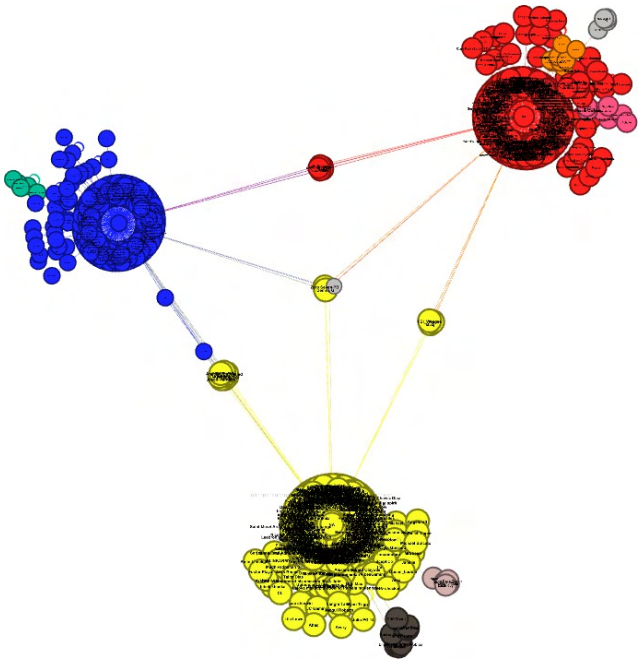
Louvain sizes as shown below, once again we can see there are 3 large communities with 400+ nodes and 16 small communities.

```
Community sizes
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19
437 443   5   2   2 446   2   2   8   4   8   2   2   2   2   2   2   5  10
>
```

The plot comes as:



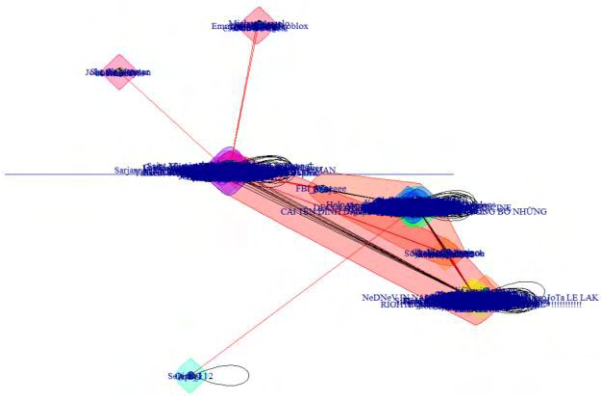
The visualization in Gephi comes as:



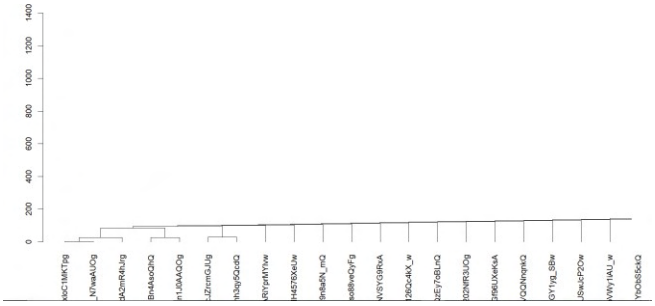
Girvan-Newman Algorithm for DJ Khaled, the community sizes are,

```
> sizes(eb_yt_actor2)
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
446   5   2   2   5 447   3   2   2   2   2   2   2  10   2   2   5   2   2   2   2   2   7   2
 25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46
  2   2   2   2   2   3   2   2 454   2   2   2   2   3   3   2   3   2   2   2   2   5
>
```

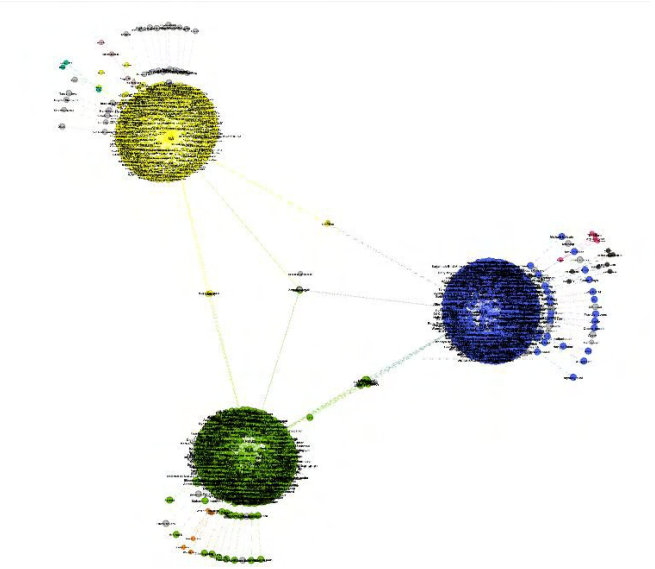
The plot comes as:



The visualization for edge-betweenness hierarchy comes as:



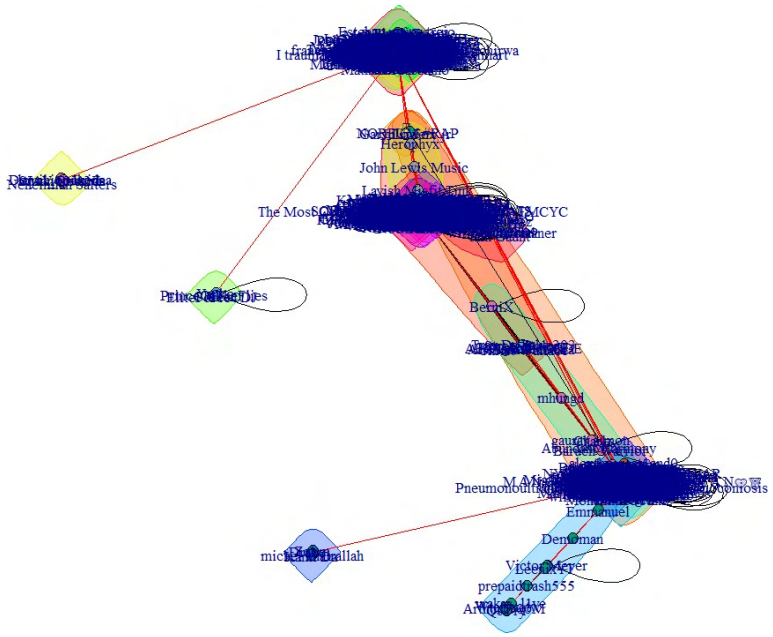
The plot in Gephi looks like:



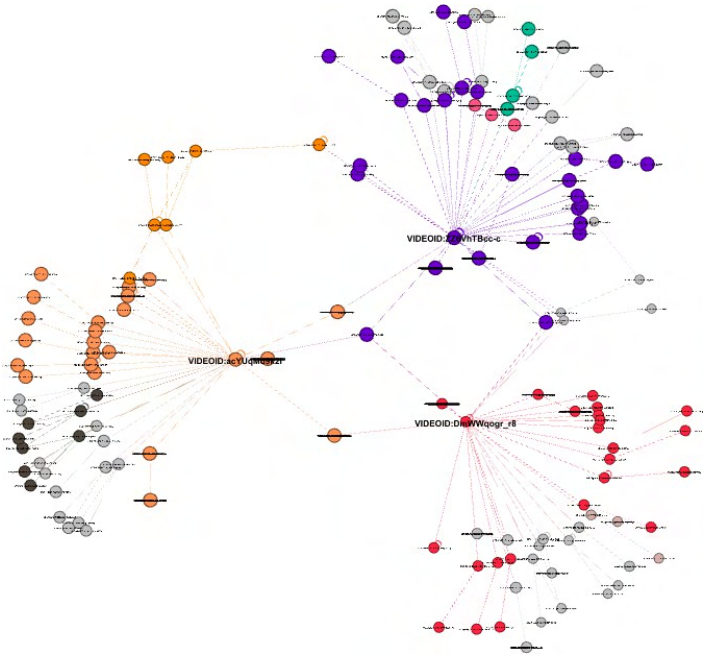
Louvain Algorithm for 21 Savage, the community sizes are,

```
> sizes(louvain_yt_actor3)
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
471 456 458   2   3   4   5   3   2   4   3   4   5   2   8   2   3   3   9   3   4   3   3   3
 25  26  27  28  29  30  31  32
  3   3   5   5   6   2   4   8
>
```

The plot comes as:



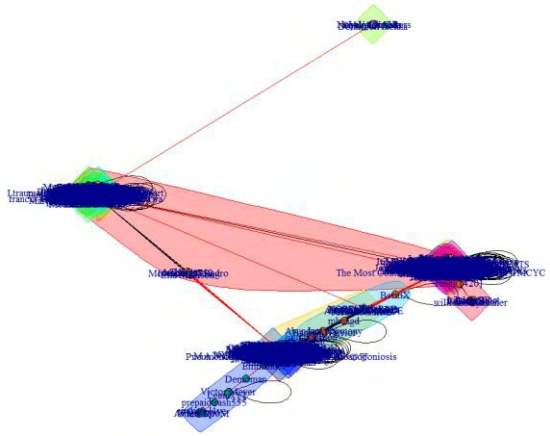
The plot in Gephi looks like:



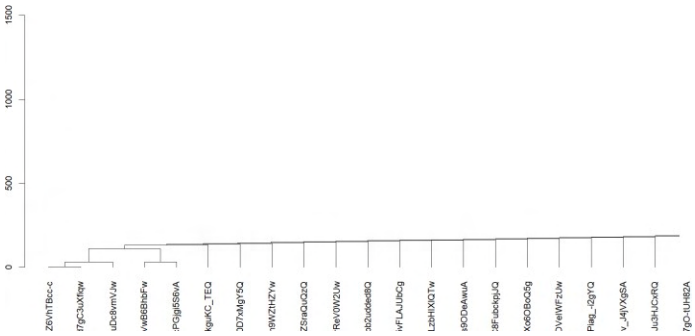
Girvan-Newman Algorithm for 21 Savage, the community sizes are,

```
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
450   2   2   2   2   2   3 451   2   4   2   2   2   2   5 451   2   2   2   4   3   2   2   2
 25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48
  4   5   2   2   2   8   2   3   2   3   9   3   4   3   2   2   3   2   3   3   2   2   3   2
 49  50  51  52  53  54
  2   2   2   4   2   8
>
```

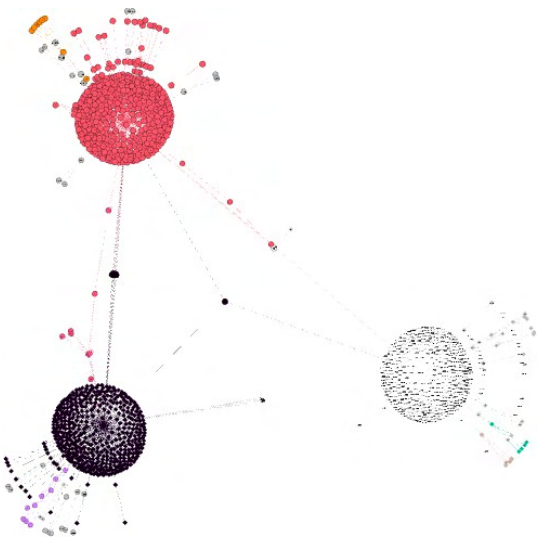
The plot comes as:



The visualization for edge-betweenness hierarchy comes as:



The plot in Gephi looks like:

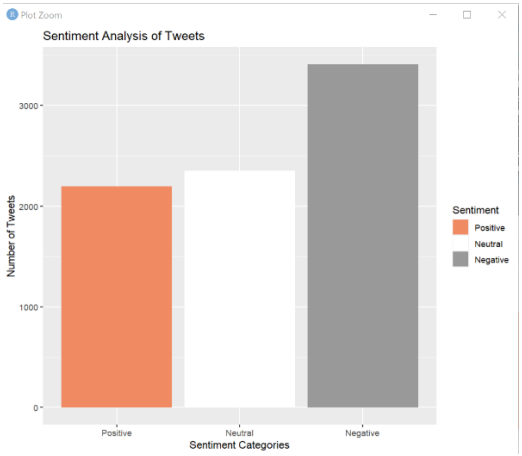


Machine Learning Models

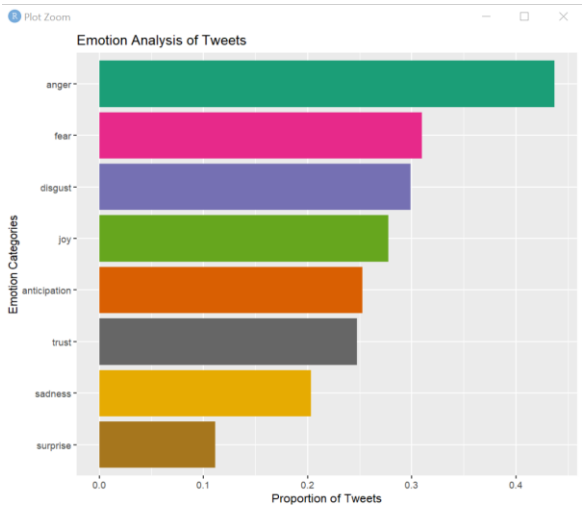
2.6) Use sentiment analysis to identify how the public reacts to events and/or topics related to your artist/band. Provide a summary of public opinions (emotions, reactions). *(=> Lab 5.2)* [1.8 marks]

ANS. I have performed sentiment analysis on 3 artists viz: Drake (The Main Artist), Justin Bieber (An artist who was 4th highest in the Degree Centrality (in) for Drake) and DJ Khaled (Related/Collaborated Artist). It was observed that Drake has very high Negative sentiment reactions while Justin had Neutral and DJ Khaled had Positive.

The sentiment scores for Drake were,



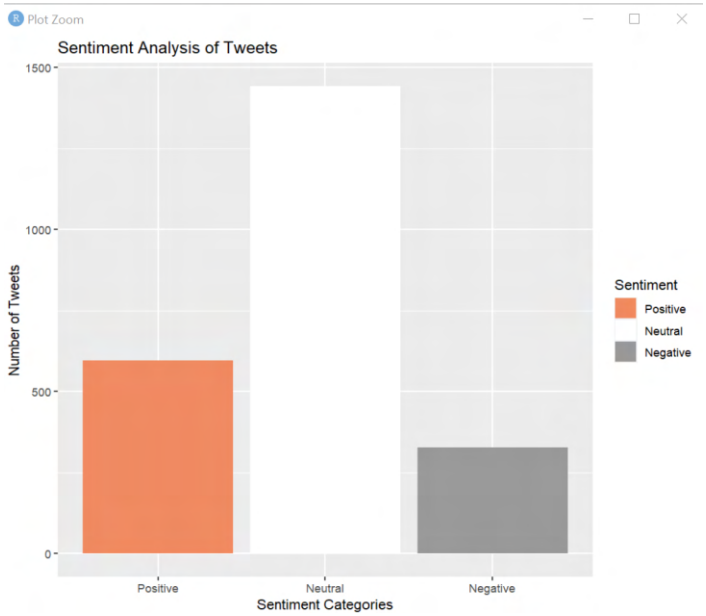
The emotional scores were,



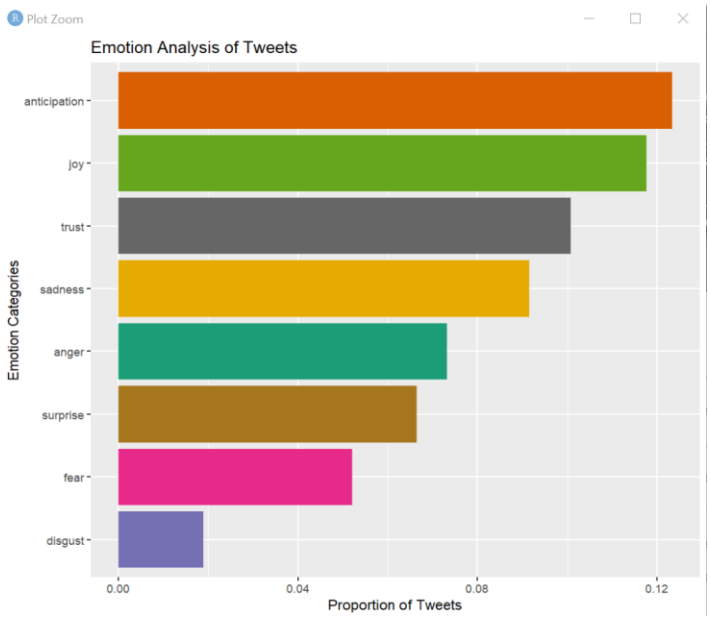
The proportion of these emotions was,

	Proportion
anger	0.4368481
fear	0.3095388
disgust	0.2991077
joy	0.2776172
anticipation	0.2526078
trust	0.2472037
sadness	0.2034686
surprise	0.1110971

Now, for Justin Bieber, The sentiment scores were,



The emotional scores were,

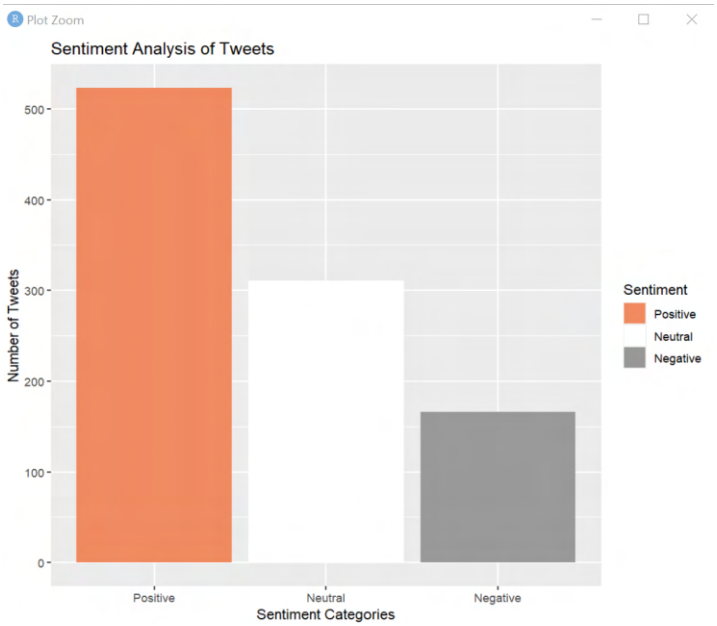


The proportion of these emotions was,

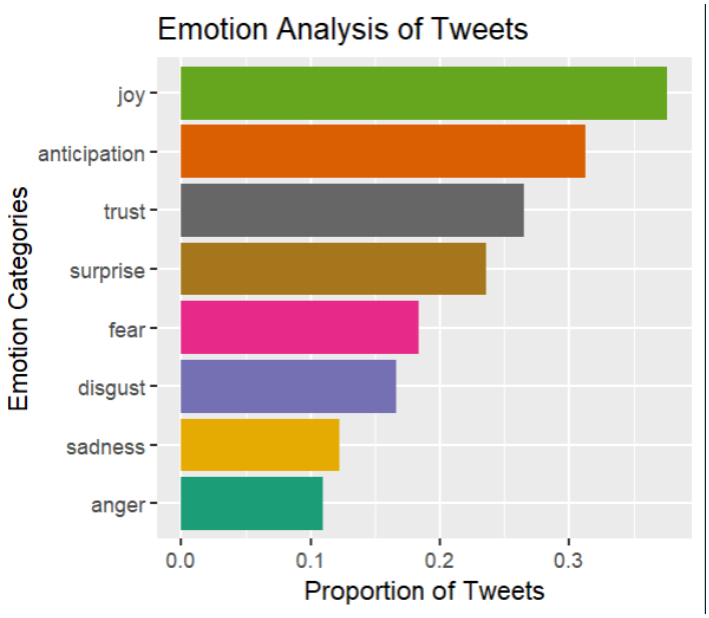
Filter

	Proportion
anticipation	0.12328767
joy	0.11763227
trust	0.10079176
sadness	0.09149177
anger	0.07314314
surprise	0.06648234
fear	0.05202966
disgust	0.01897700

Now for DJ Khaled, the sentient scores were,



The emotional scores were,



The proportion of these emotions was,

	Proportion
joy	0.376
anticipation	0.313
trust	0.265
surprise	0.236
fear	0.184
disgust	0.166
sadness	0.122
anger	0.110

2.7) Build a decision tree and evaluate its performance in predicting whether a song is by your artist/band. *(=> Lab 5.2)* [2.25 marks]

ANS. For this task, the decision tree was run through 3 different playlists of 100 songs. Those playlists are:

- Top 100 tracks currently on Spotify (Playlist ID = '4hOKQuZbraPDIfaGbM3lKI')
- Classic Road Trip Songs (Playlist ID = '37i9dQZF1DX9wC1KY45plY')
- Happy pop (Playlist ID = '37i9dQZF1DX1H4LbvY4Oji')

The performance of the Decision Trees for these playlists is as follows:

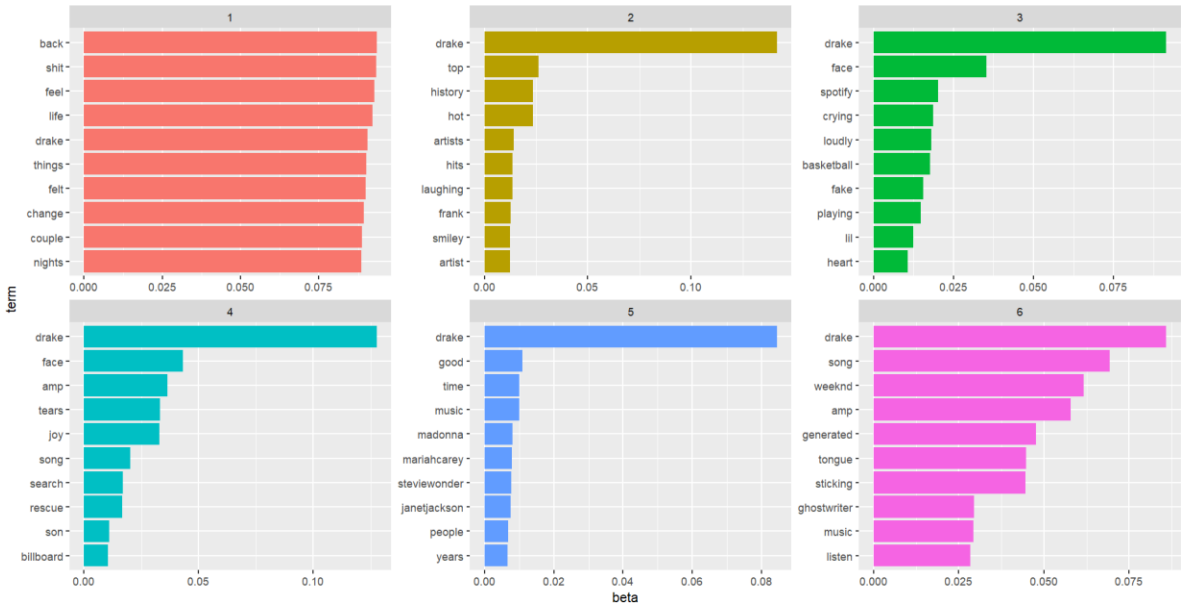
Top 100 tracks currently on Spotify	Classic Road Trip Songs	Happy pop
Reference Prediction 0 1 0 7.0 1.9 1 6.8 84.2 Accuracy (average) : 0.9123	Reference Prediction 0 1 0 9.7 1.2 1 3.2 85.9 Accuracy (average) : 0.9557	Reference Prediction 0 1 0 7.7 1.2 1 3.1 88.1 Accuracy (average) : 0.9574
Accuracy: 91.23%	**Accuracy: 95.57%**	**Accuracy: 95.74%**

The accuracy of the decision tree has improved with each iteration. The goal of the model was to classify if a song is a "Drake" song based on the audio features of the song. The first step here was to get just the audio features of Drake in a data frame. The model was then fed with songs that are not by Drake so it can train on non-Drake songs. To do that, we got audio features from the playlists mentioned above from Spotify. A class variable was added to the datasets (0 for audio features from playlists which have songs not from Drake and 1 to the audio features of Drake dataset). However, there could have been a Drake song in these playlists so anti_join() function was used to remove them from playlists and added to the Drake data set. Then randomly 80% of the records were used as training sets and remaining 20 as testing sets. The model was trained using C5.0 Decision Tree model. train() function was used to train the model and predict() was used to test the model. To check the performance of the model, a Confusion Matrix was used. The findings of the Confusion Matrix are available in the table above.

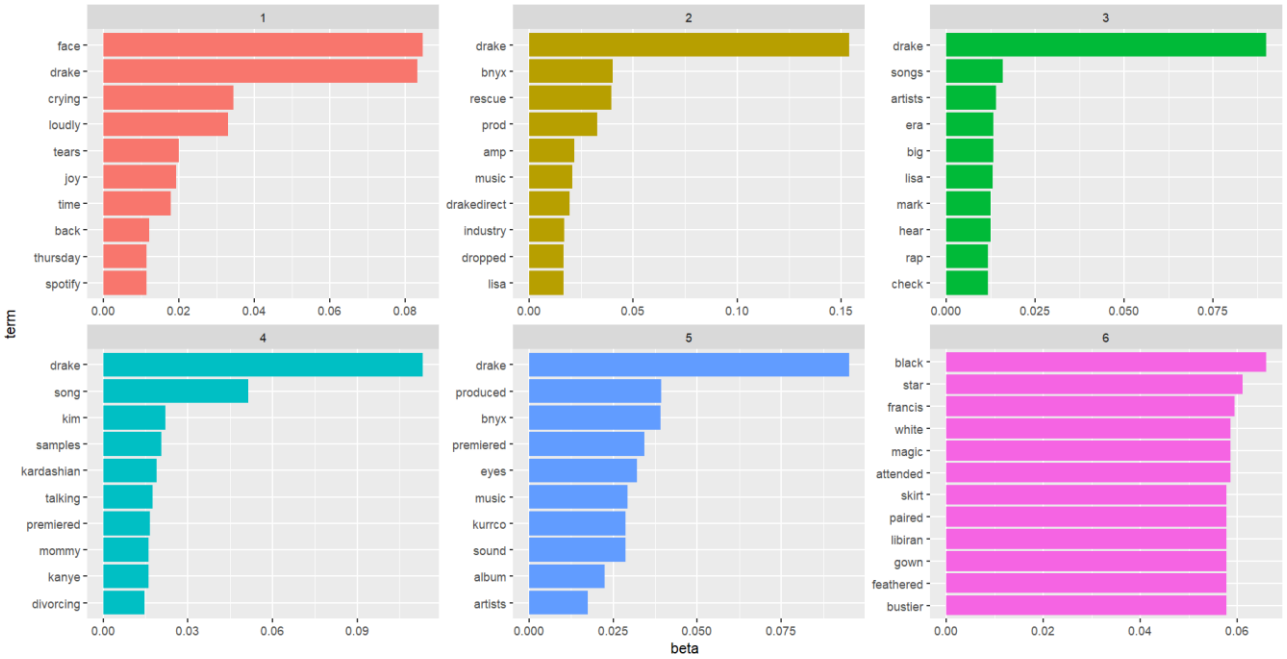
2.8) Use LDA topic modelling to identify some terms that are closely related to your artist/band. Find at least 3 significant groups of words that can be meaningful to your analysis. Explain your findings. (=> Lab 5.1) [1.8 marks]

ANS. The datasets used here are Twitter data collected for Milestone 2 and then Twitter data collected for Milestone 1.

When Milestone 2 data was used:



When Milestone 1 data was used:



LDA stands for Latent Dirichlet Allocation, is primarily used for topic modelling, which is the process of identifying topics or themes within a collection of documents. The first step is to pre- process the already retrieved data and cleaning it. Many functions from textclean package are

used to clean the data. This cleaned data is then transformed into a text corpus. A text corpus is a large and structured collection of written or spoken texts that are used for language analysis. Then, all stopwords are removed from the tweets to reduce noise in the text. A document-term matrix is made from the corpus. The null entries in the matrix are also removed. An LDA model is then created for the matrix. In my case, using k = 6, the model tried to discover 6 distinct topics within the matrix. Then, topic probability for each word is generated. This is the probability that then given word belongs to that topic. Finally, 10 words with highest probabilities for each topic is selected and visualized.

Visualisation

2.9) Visualise your Twitter actor network in Gephi, with the node size determined by the number of followers for that actor. What insights can you extract from the visualisation? (This question is a little more difficult. Skip it if you're unsure and come back later.Hint: Look at the vosonSML documentation. No further hints will be provided for the question.) [1.8 marks]

ANS. To be skipped according to announcement.

2.10) Create at least three charts from your datasets using Tableau and combine them together into a dashboard. 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. [2.25 marks]

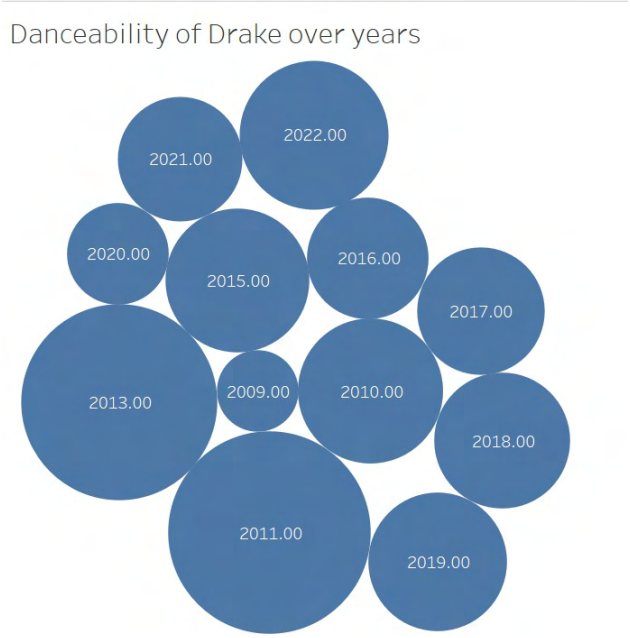
ANS. To create Tableau charts and a dashboard, I had to convert my Data frames in R to JSON files. It was done by using 'rjson' and 'jsonlite' libraries.

The following snippet was used to convert Drake Twitter data to JSON:

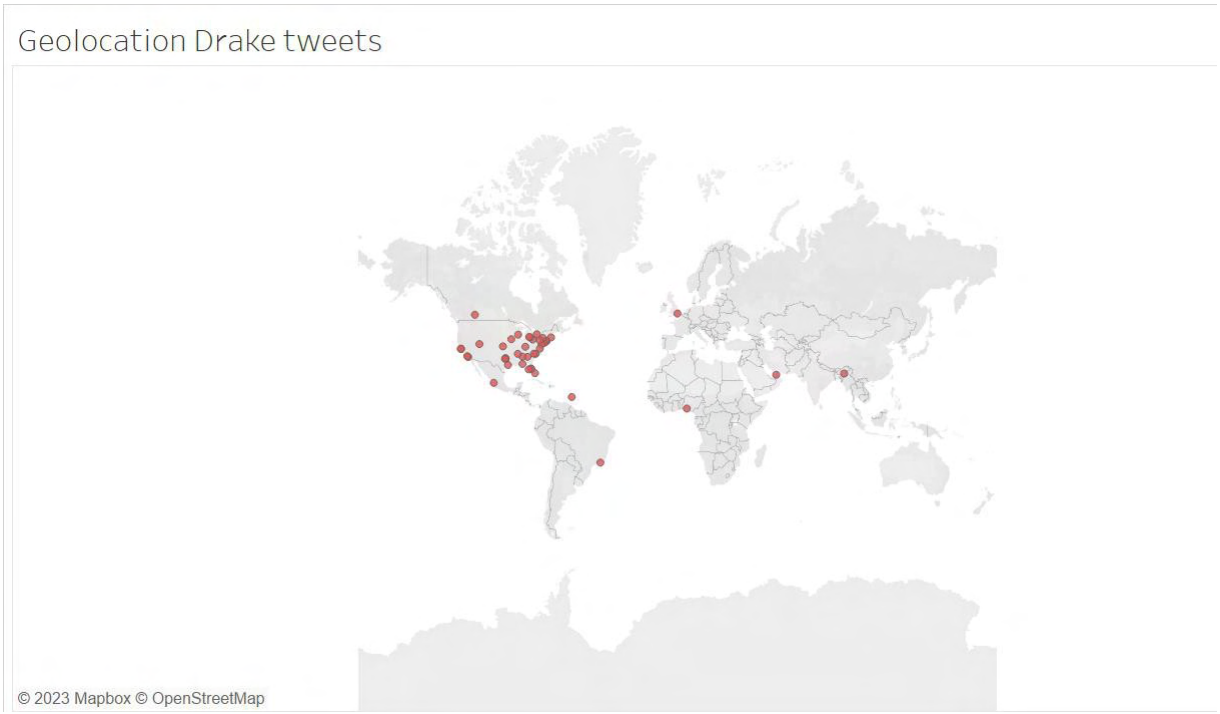
```
toJSON(x = twitter_data2, dataframe = 'values', pretty = T)

twitter_data2J <- toJSON(twitter_data2)

write(twitter_data2J, "DrakeTweets.json")
```

Some other processing was done on this data set and the Audio features data set from Spotify to get necessary data in JSON format.

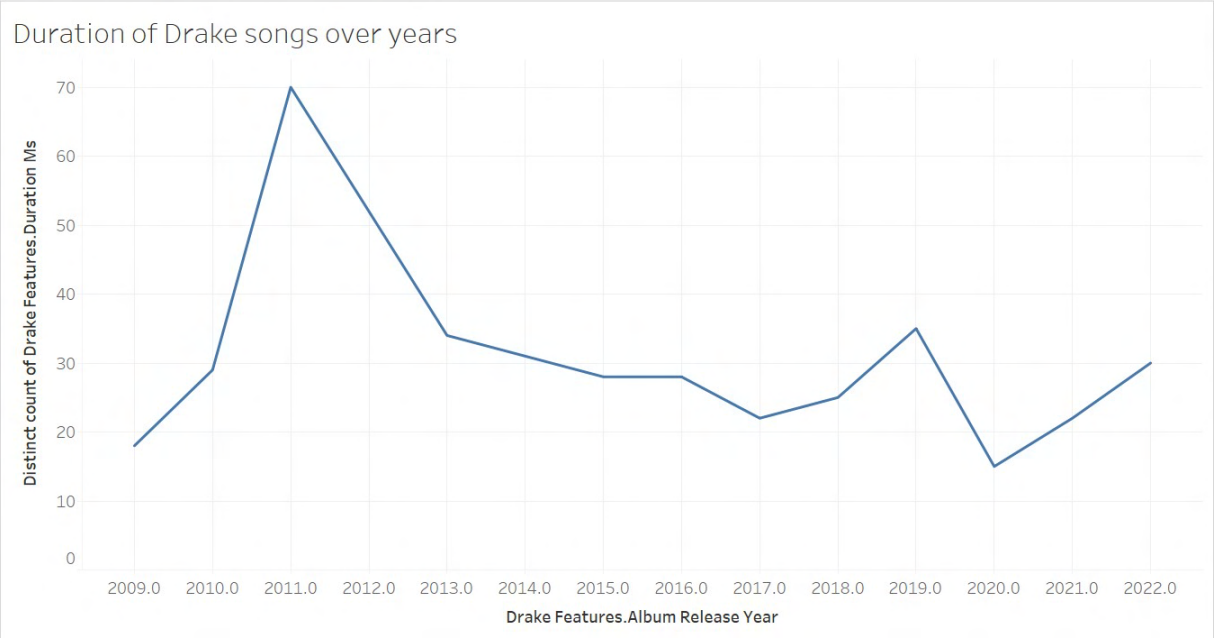
The first chart created shows the Danceability of Drake over the years. It can be observed that Danceability of Drake songs have reduced since his first album.



The next chart created was Geo location of users from the Twitter data collected on Drake.

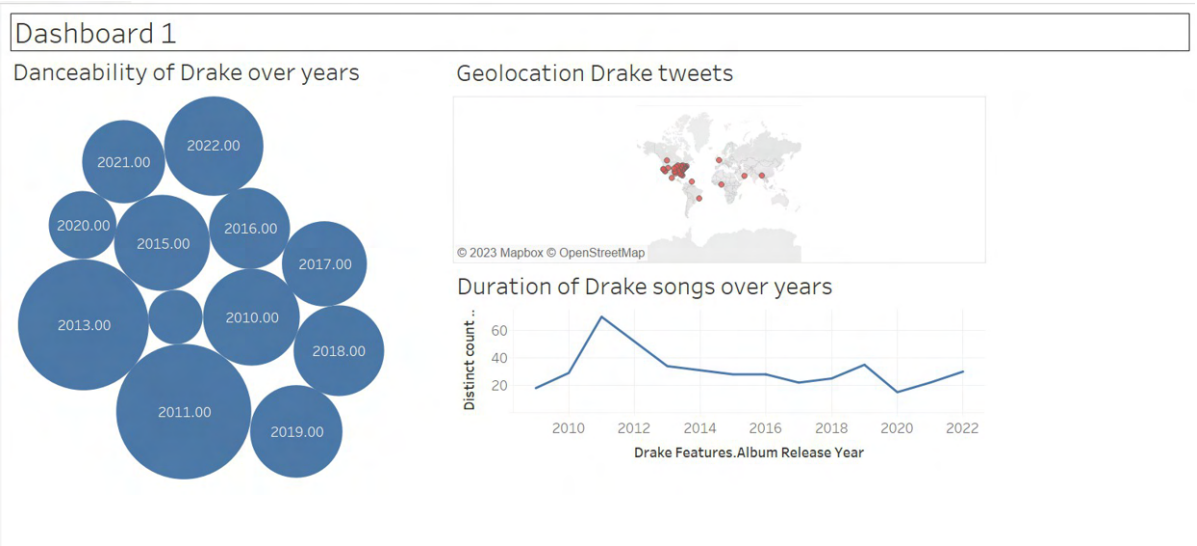


A chart showing duration of Drake songs over the years was created using the audio features data.



The duration of Drake songs has reduced since he first started making songs. However, from this plot it seems that the duration of songs has been on the rise again.

Finally, a dashboard was created using these 3 charts.



This dashboard can be accessed from Tableau Public website using the embedded link.

Analysis Review

2.11) Research and review other methods/algorithms for network analysis, machine learning models, or visualisation. Compare them to the methods you used in these milestones. 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. [2-4 paragraphs, 2.5 marks]

ANS.

We had to collect data from Twitter, Spotify, and YouTube for this milestone assignment. The Twitter data was then pre-processed before being utilised to do network analysis (centrality), sentiment analysis (Decision Trees), and topic/theme extraction (LDA). Spotify data was also utilised to collect information about artists and music, and YouTube data was used for network analysis to identify communities. Louvain and Girvan-Neuman algorithms are implemented. Following the previous analysis, plotting was done in R, Gephi, and Tableau.

Alternative methods for Network Analysis include Common Neighbours, the Jaccard Coefficient, and the Adamic/Adar Index. These techniques are used to predict missing node connectivity. The premise behind Common Neighbours is that if two nodes have a common neighbour, they are more likely to develop a link in the future. The Jaccard coefficient is derived by dividing the total number of unique neighbours of both nodes by the number of common neighbours of both nodes. The Adamic/Adar Index is based on the premise that nodes with rare or uncommon neighbours have greater accuracy in predicting for future link formation.

Other methods for machine learning that can be used are Linear Regression, Random Forest, Support Vector Regression, Naïve Bayes, Support Vector Machines, DBSCAN or Convolutional Neural Networks.

Determining which algorithms are better than Louvain, Girvan-Neuman or Decision trees was a difficult question because every algorithm has its own advantages and disadvantages.