SHEHRYAR MALLICK

s5328488

ASSIGNMENT 1

MILESTONE 2

BIG DATA ANALYTICS AND SOCIAL MEDIA 7230 ICT

s5328488

QUESTION 2.1

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)?

The following code helps in retrieving the songs and albums of the artist, “The Weeknd”.

```
# part b: How many albums and songs they have published
albums <- getAlbums("1Xyo4u8uXC1ZmMpatF05PJ", token = keys)
total_albums = n_distinct(albums$name)
total_albums

unique_albums = albums[!duplicated(albums$name), ]
View(unique_albums)

total_songs <- 0
for (i in unique_albums$id)
{

  #print(i)
  songs <- getAlbum(i, token = keys)
  total_songs <- total_songs + n_distinct(songs$name)
}
total_songs
#my_names_new <- songs[1:length(songs)]   #debugging purposes
```

The output looks like:

	id	name	album_type	available_markets
1	35dut3ICqF3NEDkjxfzJJ1	Starboy (Deluxe)	album	
3	1OARrXe5sB0gyy3MhQ8h92	Live At SoFi Stadium	album	
6	4M2Mf4pmARKGVT9MLCe3HA	Avatar: The Way of Water (Original Motion Picture Soundtra...	album	
7	12INIMsFtBjyehNnawBv36	Dawn FM (Alternate World)	album	
11	1bupWi00723vxZoS7MX9NU	Dawn FM	album	
13	6YIDIxqEjvY63ffH6AwCjd	After Hours (Deluxe)	album	
17	4yP0hdKOZPNshxUOjY0cZj	After Hours	album	
19	4qZBW3f2Q8y0k1A84d4iAO	My Dear Melancholy,	album	
21	2ODvWsOgouMbaA5xf0RkJe	Starboy	album	
23	0P3oVJBFOv3TDXIYRhGL7s	Beauty Behind The Madness	album	
25	3hhDpPtCFuQbppwYgsVhMO	Kiss Land	album	
29	3MP8mUHuQIYrGUkrEG4qpJ	Trilogy	album	
31	04hy4jb1GDD00otiwzsFUB	Echoes Of Silence (Original)	album	
32	6F87IH0I09qIrzvCCKc7Iz	Thursday (Original)	album	

The total number of albums available on spotify are 14 and total number of songs associated to our artist are 214.

```
> total_albums
[1] 14
```

```
> total_songs
[1] 214
```

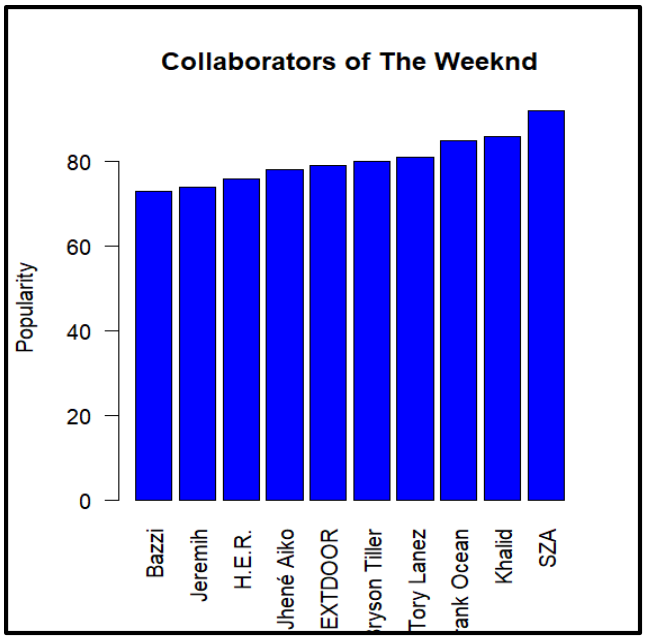
Next up I have used the following code to extract the collaborators:

```
# part c: the artist that have collaborated with the weeknd
related_artist_the_weeknd <- getRelated("The Weeknd", token = keys)
View(related_artist_the_weeknd)
related_artist_the_weeknd <- related_artist_the_weeknd[order(related_artist_the_weeknd$popularity),]
top_10_related_artist_the_weeknd <- top_n(related_artist_the_weeknd,10)

#View(top_10_related_artist_the_weeknd)

#making bar plot of the top 10 collaborators based on their popularity
barplot(top_10_related_artist_the_weeknd$popularity,
        names.arg=top_10_related_artist_the_weeknd$name,ylab="Popularity",
        col="blue",las=2,main="Collaborators of The Weeknd")

```

Which gives us the following result:

	name	id	popularity	type	followers
1	PARTYNEXTDOOR	2HPaUgqeutzr3jx5a9WyDV	79	artist	5098880
2	Roy Woods	7mDU6nMUJnOSY2Hkjz5oqM	62	artist	1038944
3	Brent Faiyaz	3tlXnStJ1fFhdScmQeLpuG	82	artist	3755130
4	Bryson Tiller	2EMAnMvWE2eb56ToJVfCWs	80	artist	6884086
5	Jeremih	3KV3p5EY4AvKxOlhGHORLg	74	artist	6290640
6	6LACK	4lVAbR2w4JJNJDDRFP3E83	77	artist	4182047
7	Majid Jordan	4HzKw8XcD0piJmDrrPRCYk	67	artist	862523
8	Daniel Caesar	20wkVLutqVOYrc0kxFs7rA	82	artist	4038438
9	Miguel	360lAlyVv4PCEVjgyMZrxK	82	artist	4619536
10	Tory Lanez	2jku7tDXc6XoB6MO2hFuqg	81	artist	5445825
11	Khalid	6LuN9FCkKOj5PcnpouEgny	86	artist	15947533
12	Bazzi	4GvEc3ANtPPjt1ZJIIr5ZI	73	artist	5406810
13	Summer Walker	57LYzLEk2LcFghVwuWbcuS	80	artist	4933667
14	Chase Atlantic	7cYEt1pqMgXJdq00hAwVpT	80	artist	4845975
15	Jhené Aiko	5ZS223C6JyBfXasXxrRqOk	78	artist	6421328
16	SZA	7tYKF4w9nC0nq9CsPZTHyP	92	artist	12264179
17	H.E.R.	3Y7RZ31TRPVadSFVy1o8os	76	artist	6299073
18	Frank Ocean	2h93pZq0e7k5yf4dywlkpM	85	artist	11048590
19	Don Toliver	4Gso3d4CscCijv0ImajZWs	84	artist	2835677
20	NAV	7rkW85dBwwrJtlHRDkJDAC	75	artist	3290308

To better understand the result I have extracted the top 10 collaborators of The Weeknd on the basis of popularity.

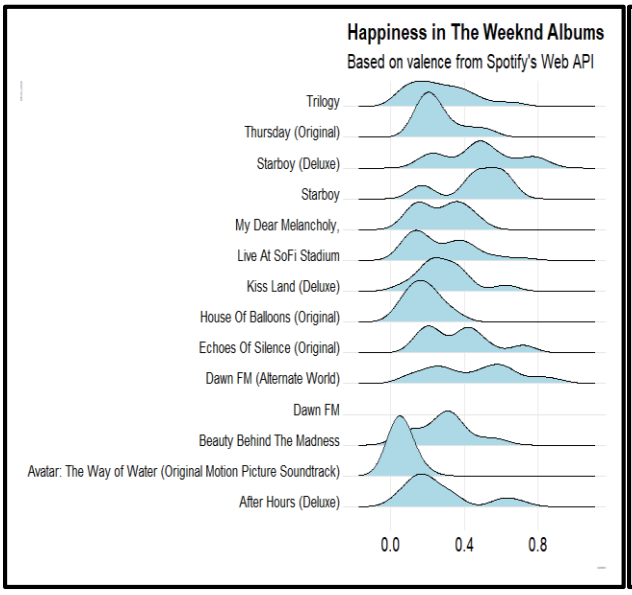


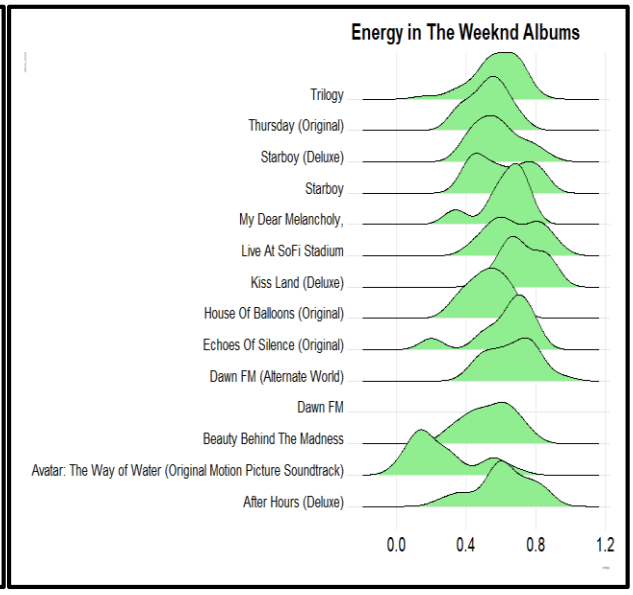
Finally we have calculated the prevalent features of our artist. The features that I have selected for the purpose of the assessment are:

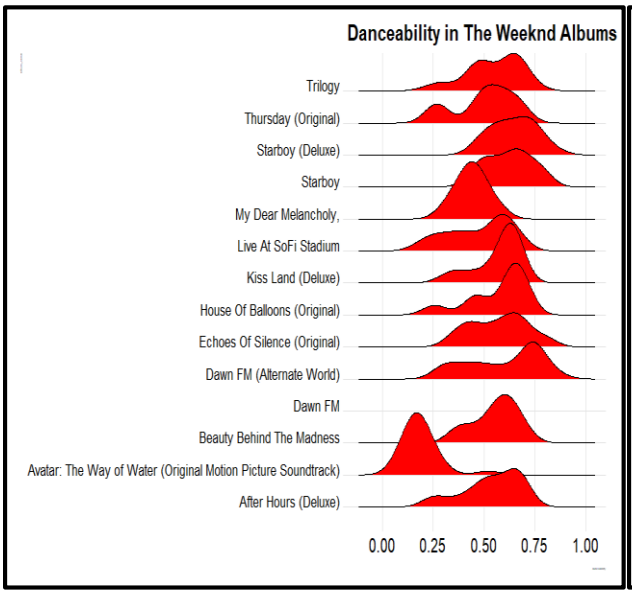
1) Happiness (valence)
2) Energy
3) Danceability
4) Speechiness

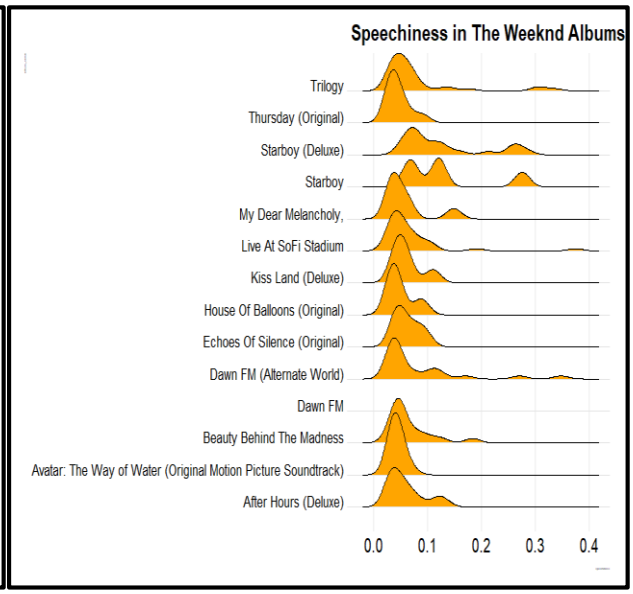
```
# Part d: prevalent features the weeknd
audio_features <- get_artist_audio_features("The Weeknd")
View(audio_features)
audio_features <- audio_features[!duplicated(audio_features$track_name), ]

# Plot happiness (valence) scores for each album
ggplot(audio_features, aes(x = valence, y = album_name)) +
  geom_density_ridges(fill = "lightblue") +
  theme_ridges() + theme(axis.title = element_text(size = 1.5)) +
  theme(axis.text.y = element_text(size = 10)) +
  ggtitle("Happiness in The Weeknd Albums",
          subtitle = "Based on valence from Spotify's Web API")

# Plot energy scores for each album
ggplot(audio_features, aes(x = energy, y = album_name)) +
  geom_density_ridges(fill = "lightgreen") +
  theme_ridges() + theme(axis.title = element_text(size = 1.5)) +
  theme(axis.text.y = element_text(size = 10)) +
  ggtitle("Energy in The Weeknd Albums")

# Plot danceability scores for each album
ggplot(audio_features, aes(x = danceability, y = album_name)) +
  geom_density_ridges(fill = "red") +
  theme_ridges() + theme(axis.title = element_text(size = 1.5)) +
  theme(axis.text.y = element_text(size = 10)) +
  ggtitle("Danceability in The Weeknd Albums")

# Plot speechiness scores for each album
ggplot(audio_features, aes(x = speechiness, y = album_name)) +
  geom_density_ridges(fill = "orange") +
  theme_ridges() + theme(axis.title = element_text(size = 1.5)) +
  theme(axis.text.y = element_text(size = 10)) +
  ggtitle("Speechiness in The Weeknd Albums")
```

The code yields the following results:









While comparing the data retrieved in this step and the one conducted in step 1.1 of milestone 1 the only significant finding worth mentioning is the number of albums, as per findings here there are a total of 14 albums as for the data gathered in contrast to this indicated a total of 16 albums.

QUESTION 2.2

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.)

We have used the following code to extract YouTube videos related to The Weeknd, however due to constraints we have extracted data relevant to 10 videos only.

```
155  yt_oauth(app_id = client_id, app_secret = client_secret, token = '')
156
157  video_search <- yt_search("The Weeknd")
158  View(video_search)
```

Following is the data that we have extracted.

	video_id	publishedAt	channelId	title
1	dMoFcvfd5t4	2023-04-21T04:00:15Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd ft. Future - Double Fantasy (Official Music Video)
2	XXYlFuWEuKI	2021-01-05T17:00:12Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd - Save Your Tears (Official Music Video)
3	YQ-qToZUybM	2023-02-24T05:00:18Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd & Ariana Grande - Die For You (Remix) (Of...
4	4NRXx6U8ABQ	2020-01-21T18:00:10Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd - Blinding Lights (Official Video)
5	u9n7Cw-4_HQ	2021-10-22T04:00:11Z	UCF_fDSgPpBQuh1MsUTgIARQ	Swedish House Mafia and The Weeknd - Moth To A Flame (...
6	34Na4j8AVgA	2016-09-28T16:00:01Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd - Starboy ft. Daft Punk (Official Video)
7	2fDzCWNS3ig	2022-04-05T13:00:10Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd - Out of Time (Official Video)
8	JZjAg6fK-BQ	2017-02-16T17:00:03Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd - Reminder (Official Video)
9	CVw7iulcl98	2022-07-04T12:45:31Z	UC-AlofdKECUdhXrbJQZ6iEg	The Weeknd - DIE FOR YOU (Lyrics)
10	yzTuBuRdAyA	2015-05-27T13:00:03Z	UCF_fDSgPpBQuh1MsUTgIARQ	The Weeknd - The Hills

Then we looped through each of the songs and extracted the data relevant to each video and stored in a dataframe.

```
160  #looping through the first 10 videos to get the stats
161  the_weeknd_first_10_videos_search <- data.frame()
162
163  for (i in 1:10)
164  {
165    the_weeknd_first_10_videos_search <- rbind(the_weeknd_first_10_videos_search,
166                                               get_stats(video_id = video_search$video_id[i]))
167  }
168
169  View(the_weeknd_first_10_videos_search)
```

The output looks like following:

	id	viewCount	likeCount	favoriteCount	commentCount
1	dMoFcvfd5t4	9952667	388569	0	11499
2	XXYlFuWEuKI	1166005020	7865524	0	280920
3	YQ-qToZUybM	76784610	1412151	0	34415
4	4NRXx6U8ABQ	709695788	9118343	0	285603
5	u9n7Cw-4_HQ	75166309	1072249	0	31374
6	34Na4j8AVgA	2231919062	10965288	0	390722
7	2fDzCWNS3ig	87181250	1568600	0	41733
8	JZjAg6fK-BQ	411250796	2891391	0	71377
9	CVw7iulcl98	20050635	188869	0	2914
10	yzTuBuRdAyA	1973878062	12017149	0	309223

The following video having id: has the most likes count

```
> the_weeknd_first_10_videos_search[which.max(the_weeknd_first_10_videos_search$likeCount),]
           id  viewCount likeCount favoriteCount commentCount
10 yzTuBuRdAyA 1973878062  12017149             0       309223
```

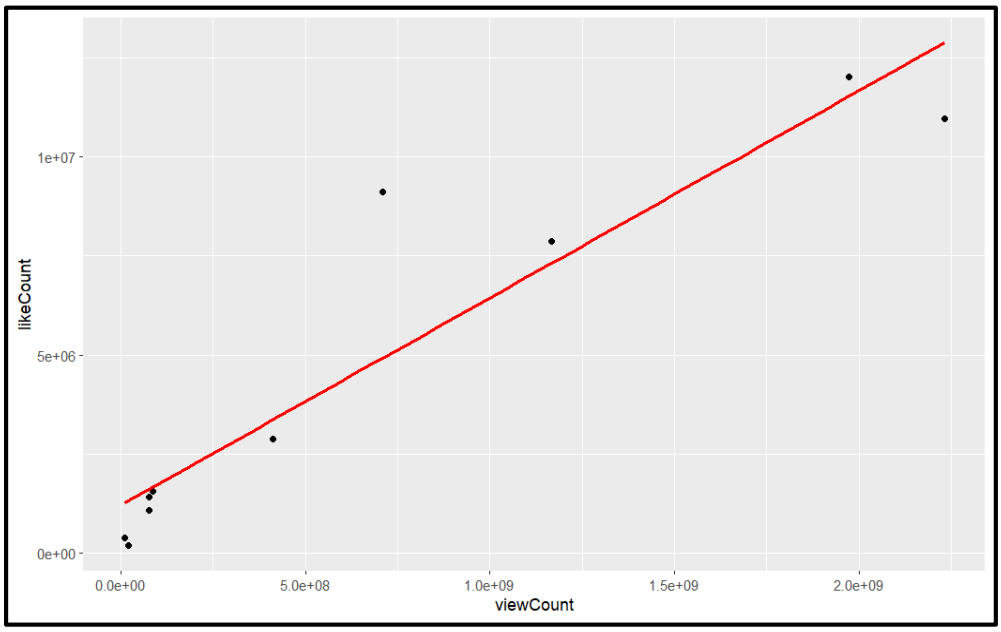
The following video having id: has the most view count:

```
> the_weeknd_first_10_videos_search[which.max(the_weeknd_first_10_videos_search$viewCount),]
           id  viewCount likeCount favoriteCount commentCount
6 34Na4j8AVgA 2231919062  10965288             0       390722
```

After that we analyzed if there exist any relation between the views on the video and the likes it got.

```
# analyzing correlation between views and likes
the_weeknd_first_10_videos_search <-
  transform(the_weeknd_first_10_videos_search, likeCount = as.numeric(likeCount),
                                               viewCount = as.numeric(viewCount))

view_likes_df <- the_weeknd_first_10_videos_search[order(the_weeknd_first_10_videos_search$viewCount)

#View(view_likes_df) #for debugging

ggplot(view_likes_df, aes(x=viewCount, y=likeCount)) +
  geom_point() +
  geom_smooth(method=lm , color="red", se=FALSE)
```

To observe the relation I also plotted a regression line as it helps to see if there exists a correlation between the data or not.



Upon reviewing the above graph it can be said with certainty that there exists a **positive correlation** between **view count and like count** on YouTube.

QUESTION 2.3

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)?

Following images shows the extraction of tweets for this question.

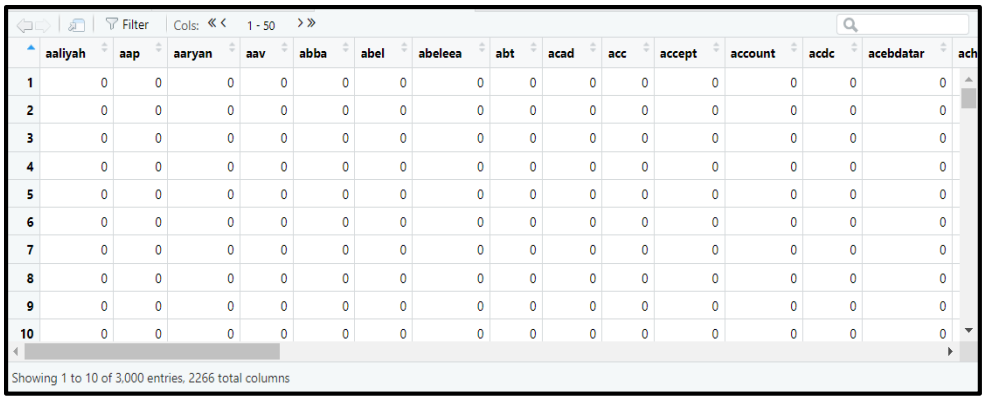
```
Collecting tweets for search query...
Search term: The Weeknd
Requested 5000 tweets of 3000 in this search rate limit.
Rate limit reset: 2023-05-04 00:56:00

tweet        | status_id           | created
-------------------------------------------------------
Latest Obs   | 1653924395897716737 | 2023-05-04 00:47:59
Earliest Obs | 1653819568668954627 | 2023-05-03 17:51:26
Collected 3000 tweets.
RDS file written: 2023-05-04_005710-TwitterData.rds
Done.
>
```

The following code is used to preprocess the tweets and then develop a term frequency term matrix of the collected tweets.

```
# Clean the tweet text
clean_text <- the_weeknd_twitter_data$tweets$text %>%
  rm_twitter_url() %>%
  replace_url() %>%
  replace_hash() %>%
  replace_tag() %>%
  replace_emoji() %>%
  replace_emoticon()

# Convert clean_text vector into a document corpus (collection of documents)
text_corpus <- VCorpus(VectorSource(clean_text))

# Perform further pre-processing
text_corpus <- text_corpus %>%
  tm_map(content_transformer(tolower)) %>%
  tm_map(removeNumbers) %>%
  tm_map(removePunctuation) %>%
  tm_map(removeWords, stopwords(kind = "SMART")) %>%
  tm_map(stemDocument) %>%
  tm_map(stripWhitespace)

#text_corpus[[1]]$content  #to view the preprocessed tweet text
```

```
# Transform corpus into a Document Term Matrix
the_weeknd_doc_term_matrix <- DocumentTermMatrix(text_corpus)

# Sort words by total frequency across all documents
the_weeknd_dtm_df <- as.data.frame(as.matrix(the_weeknd_doc_term_matrix))
View(the_weeknd_dtm_df)
```

The document frequency matrix looks like the following:

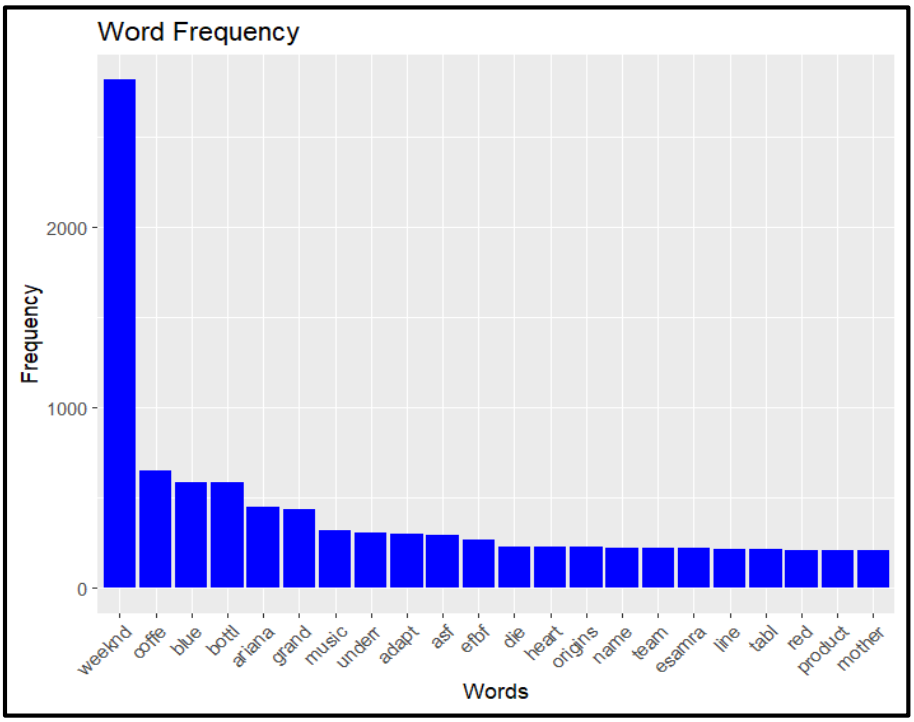


	aaliyah	aap	aaryan	aav	abba	abel	abeleea	abt	acad	acc	accept	account	acdc	acebdatar	ach
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Showing 1 to 10 of 3,000 entries, 2266 total columns

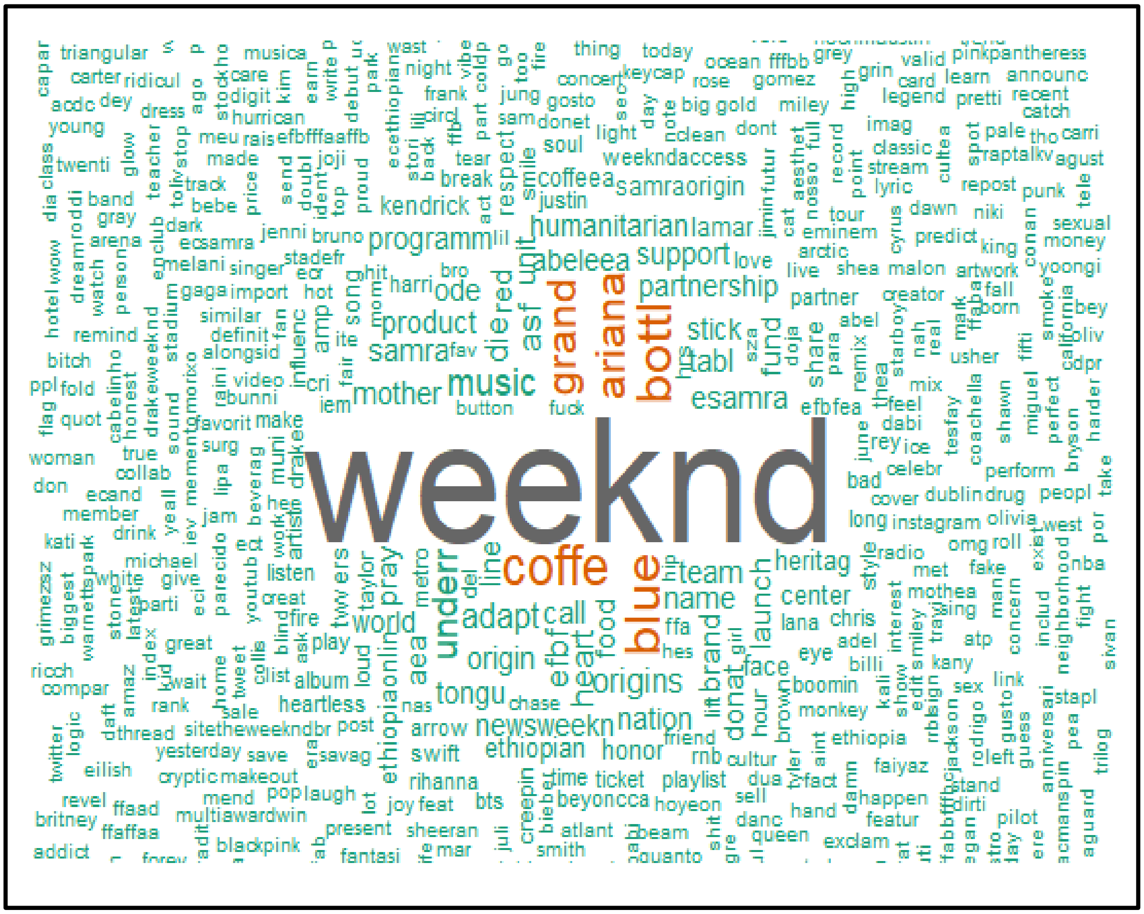
Then I plotted the words who's frequency is greater than 200

```
213 #plotting all the words that have frequency greater that 200
214 ggplot(subset(word_frequ_df, freq > 200), aes(x = reorder(word, -freq), y = freq)) +
215   geom_bar(stat = "identity", fill = "blue") +
216   theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
217   ggtitle("Word Frequency") +
218   xlab("Words") +
219   ylab("Frequency")
```



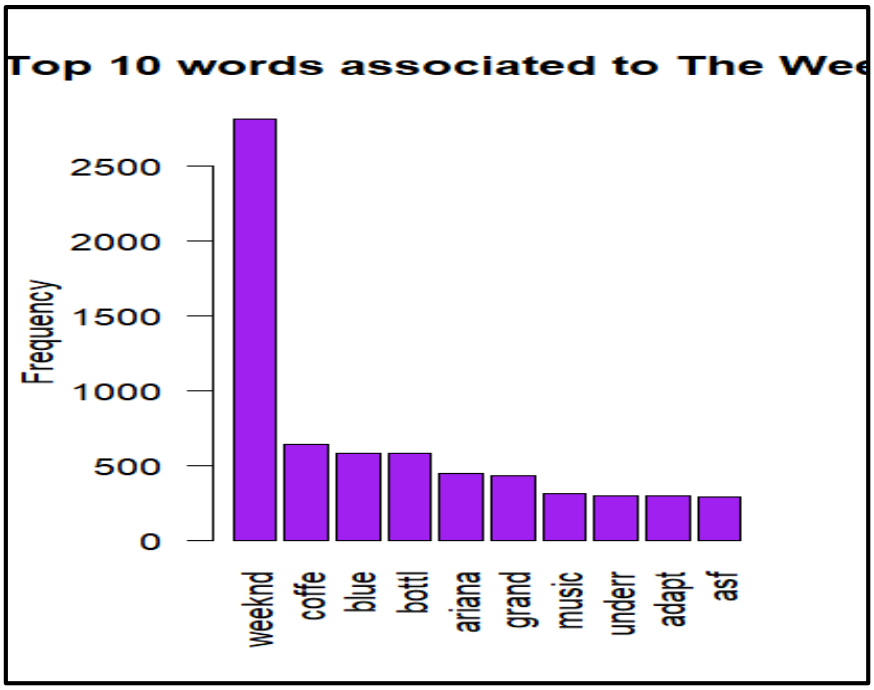
Finally we made a word cloud for all the frequent words, having frequency greater than 1 and max number of words in the word cloud to be 1000.

```r
# making wordcloud of 1000 words having frequency greater than 1
wordcloud(words=word_frequ_df$word, freq=word_frequ_df$freq, min.freq=1,
          max.words=1000, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```



Then we plotted the top 10 used words in our dataset.

```r
top_10_words <- top_n(word_frequ_df,10)

barplot(top_10_words$freq,names.arg=top_10_words$word,ylab="Frequency",
        col="purple",las=2,main="Top 10 words associated to The Weeknd")
```



In comparison to the data of 1.4 milestone 1 the top 10 frequent words were

```
> head(rank_twitter_semantic_the_weeknd, n = 10)
        #nowplaying         #metroboomin          #afterhours       #arianagrande
        0.012726266         0.010566705         0.010336954         0.009977044
           21savage               remix              #music         #taylorswift
        0.008600212         0.006930872         0.006876972         0.006443432
        #creepin #guinnessworldrecords
        0.005665335         0.005545086
```

Which were calculated using the page rank algorithm as compared to the current data that was calculated using the document frequency matrix.

QUESTION 2.4

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 are 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.

To perform the centrality analysis I have used the following code:

```
# Create twomode (bimodal) network
the_weeknd_twomode_network <- the_weeknd_twitter_data %>% Create("twomode",
                                                removeTermsOrHashtags = c("#TheWeeknd"))
the_weeknd_twomode_graph <- the_weeknd_twomode_network %>% Graph()


# Write graph to file
write.graph(the_weeknd_twomode_graph, file = "TheWeekndTwitterTwomode.graphml",
            format = "graphml")

```

```
# Find all maximum components that are weakly connected
the_weeknd_twomode_comps <- components(the_weeknd_twomode_graph, mode = c("weak"))

the_weeknd_twomode_comps$no    #how many components (island) our network has
```

```
> the_weeknd_twomode_comps$no    #how many components (island) our network has
[1] 170
```

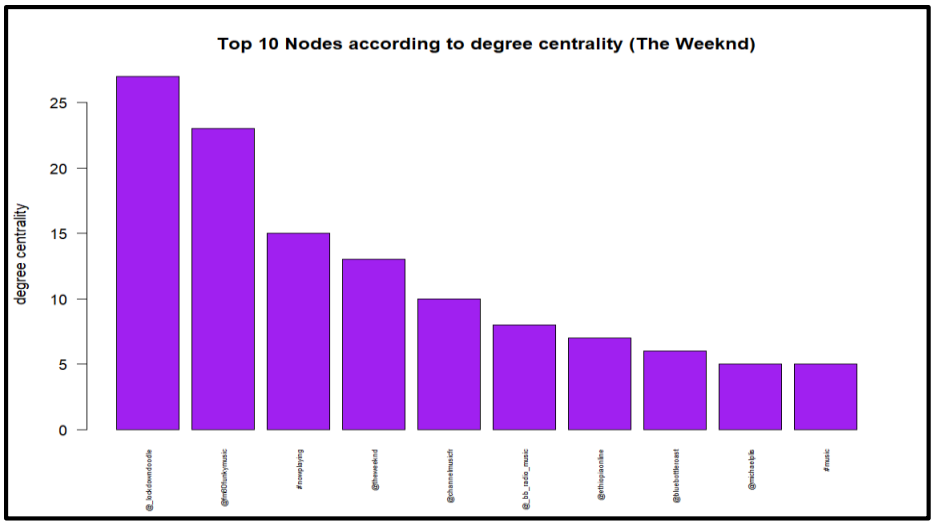
```
the_weeknd_twomode_comps$csize #size of each island
```

```
> the_weeknd_twomode_comps$csize #size of each island
  [1]  14 112   5  12   2   2   5   2   3   6  12  18   2   4   2   2   7   2   3   5   3   3
 [23]   5  21   4   2   2   2   2   2   4   2   4   2   2   2   2   2   2   5   2   2   2   2
 [45]   4   2   2   2   2   5   2   2   2   2   4   2   2   3   3   2   4   2   2   2   2   2
 [67]   3   2   3   2   3   2   2   6   2   2   2   3   2   2   2   2   4   2   3   3   2   2
 [89]   2   2   2   2   2   3   2   4   3   2   2   2   2   2   2   2   4   2   2   2   2   2
[111]   3   2   2   2   2   2   2   3   2   3   3   3   2   2   7   4   2   3   2   2   2   3
[133]   2   2   2   3   3   2   2   5   2   2   2   2   3   3   4   2   2   2   6   2   2   3
[155]   2   2   3   2   2   3   2   2   2   2   2   2   2   2   2   2
```

After which I have extracted the top 10 nodes on the basis of degree centrality

```
# Display top 10 nodes from the sub-graph ordered by degree centrality with total mode
the_weeknd_top10_nodes_degree_centrality <- data.frame(sort(degree(twomode_subgraph, mode = "total"), decreasing = TRUE)[1:10])
colnames(the_weeknd_top10_nodes_degree_centrality) <- c('Degree Centrality')
View(the_weeknd_top10_nodes_degree_centrality)
barplot(the_weeknd_top10_nodes_degree_centrality$`Degree Centrality`,
        names.arg=row.names(the_weeknd_top10_nodes_degree_centrality),ylab="Frequency",
        col="purple",cex.names=0.5,las=2,main="Top 10 Nodes according to degree centrality (The Weeknd)")

```

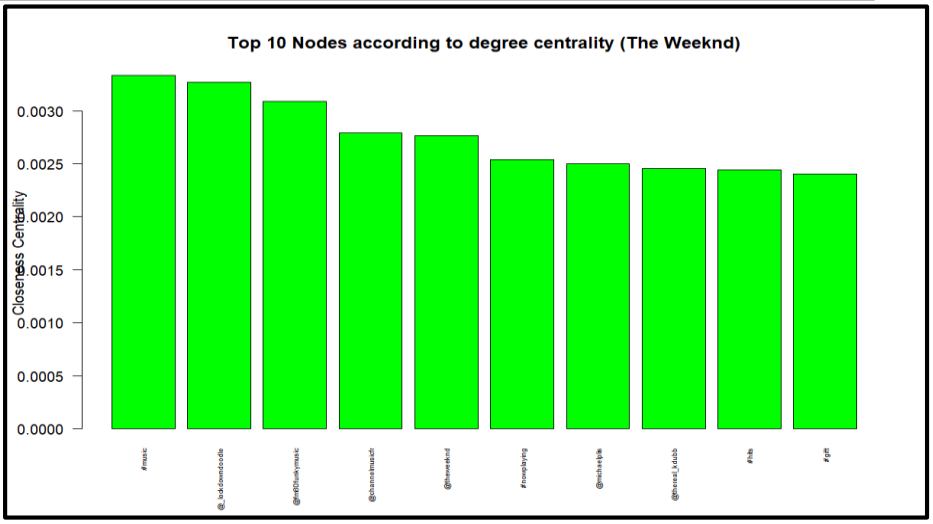
	Degree Centrality
@_lockdowndoodle	27
@fm80funkymusic	23
#nowplaying	15
@theweeknd	13
@channelmusicfr	10
@_bb_radio_music	8
@ethiopiaonline	7
@bluebottleroast	6
@michaelplis	5
#music	5



Now on the basis of closeness centrality:

```
# Display top 10 nodes from the sub-graph ordered by closeness centrality with total mode
the_weeknd_top10_nodes_closeness_centrality <- data.frame(sort(closeness(twomode_subgraph, mode = "total"), decreasing = TRUE)[1:10])
colnames(the_weeknd_top10_nodes_closeness_centrality) <- c('Closeness Centrality')
View(the_weeknd_top10_nodes_closeness_centrality)
barplot(the_weeknd_top10_nodes_closeness_centrality$`Closeness Centrality`,
        names.arg=row.names(the_weeknd_top10_nodes_closeness_centrality),ylab="Closeness Centrality",
        col="green",cex.names=0.5,las=2,main="Top 10 Nodes according to degree centrality (The Weeknd)")

```

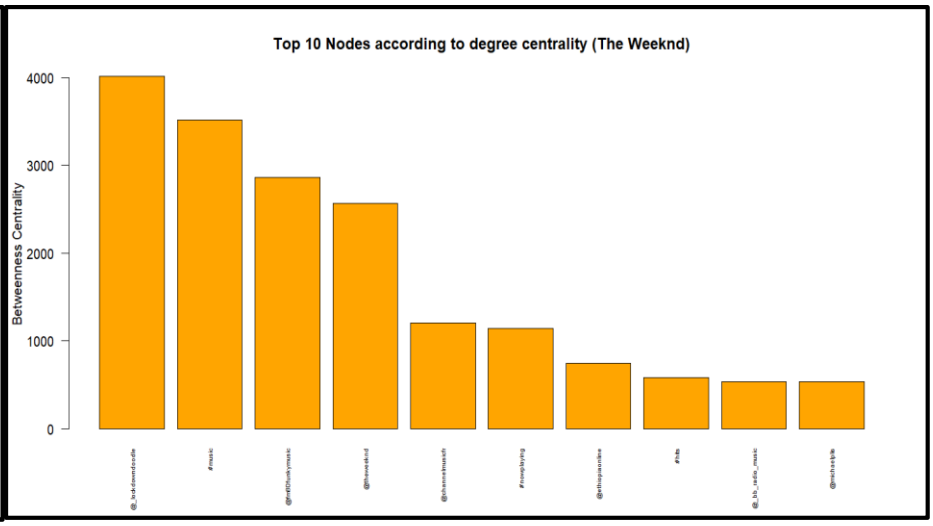
	Closeness Centrality
#music	0.003333333
@_lockdowndoodle	0.003267974
@fm80funkymusic	0.003086420
@channelmusicfr	0.002793296
@theweeknd	0.002762431
#nowplaying	0.002538071
@michaelplis	0.002500000
@thereal_kdubb	0.002450980
#hits	0.002439024
#gift	0.002403846



Finally on the basis of betweenness:

```
# Display top 10 nodes from the sub-graph ordered by betweenness
the_weeknd_top10_nodes_betweenness <- data.frame(sort(betweenness(twomode_subgraph, directed = FALSE),
                                                      decreasing = TRUE)[1:10])
colnames(the_weeknd_top10_nodes_betweenness) <- c('Betweenness Centrality')
View(the_weeknd_top10_nodes_betweenness)
barplot(the_weeknd_top10_nodes_betweenness$`Betweenness Centrality`,
        names.arg=row.names(the_weeknd_top10_nodes_betweenness),ylab="Betweenness Centrality",
        col="orange",cex.names=0.5,las=2,main="Top 10 Nodes according to degree centrality (The Weeknd)")

```

	Betweenness Centrality
@_lockdowndoodle	4016.000
#music	3513.600
@fm80funkymusic	2865.333
@theweeknd	2565.000
@channelmusicfr	1205.667
#nowplaying	1143.433
@ethiopiaonline	746.000
#hits	585.600
@_bb_radio_music	540.000
@michaelplis	539.000



The related artist that I have chosen for comparison is Ariana Grande.

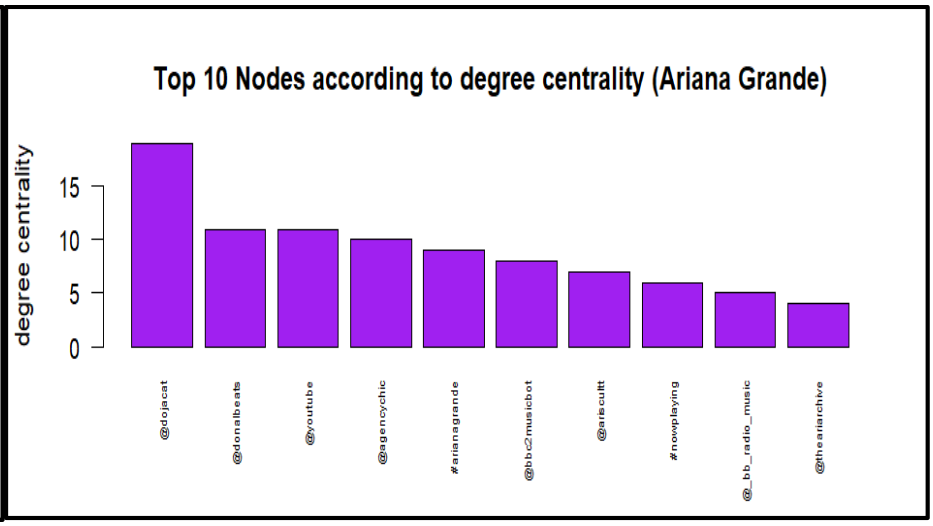
```
Collecting tweets for search query...
Search term: Ariana Grande
Requested 5000 tweets of 18000 in this search rate limit.
Rate limit reset: 2023-05-05 00:21:10

tweet        | status_id           | created
-------------------------------------------------------
Latest Obs   | 1654276235105632257 | 2023-05-05 00:06:04
Earliest Obs | 1654034439566168065 | 2023-05-04 08:05:15
Collected 4921 tweets.
RDS file written: 2023-05-05_000758-TwitterData.rds
Done.
```

```
> ariana_grande_twomode_comps$no   #how many components (island) our network has
[1] 165
```

```
> ariana_grande_twomode_comps$csize #size of each island
  [1]  7  7  3  2  3  2  2  2  2  2  2 98  2 11  2  4  3  3  2  2  2  2  2  3  2  2  2  2  2  2  4 10  2  2
 [35]  2  2  2  2  3  2  3  5  7  2  6  2  5  7  3  2  2  3  2  6  2  3  7  3  4  2  3  2  3  3  5  2  2  2
 [69]  2  2  3  3  2  5  2  2  3  2  4  2  4  5  2  4  3  2  4  2  3  2  2  2  2  5  8  4  5  2  3  4  6  2
[103]  2  2  2  5  4  3  3  3  2  3  2  3  2  3  4  2  2  2  2  2  3  2  2  2  2  2  2  2  2  2  2  4  3
[137]  2  2  2  2  2  2  2  3  2  3  2  3  3  2  2  4  2  3  3  2  2  2  2  2  2  6  2  2  2
```

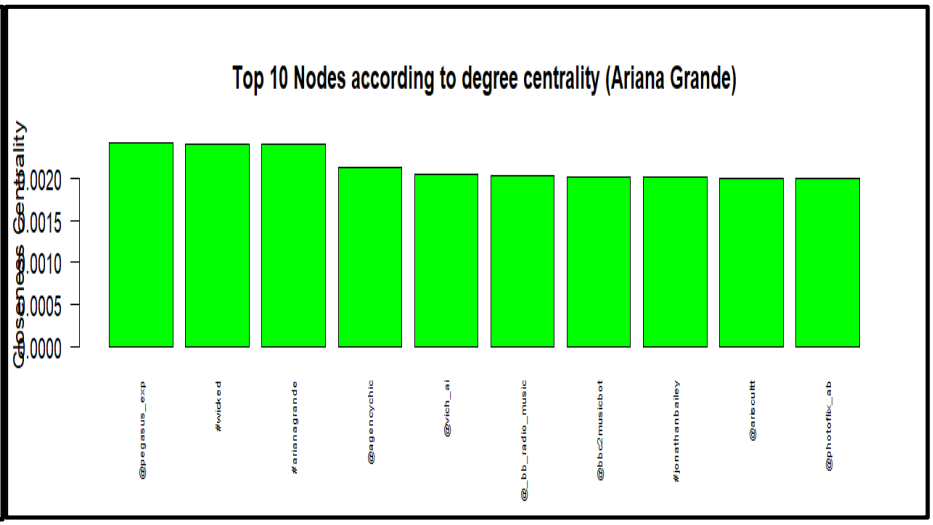
Degree centrality:

	Degree Centrality
@dojacat	19
@donalbeats	11
@youtube	11
@agencychic	10
#arianagrande	9
@bbc2musicbot	8
@ariscultt	7
#nowplaying	6
@_bb_radio_music	5
@theariarchive	4



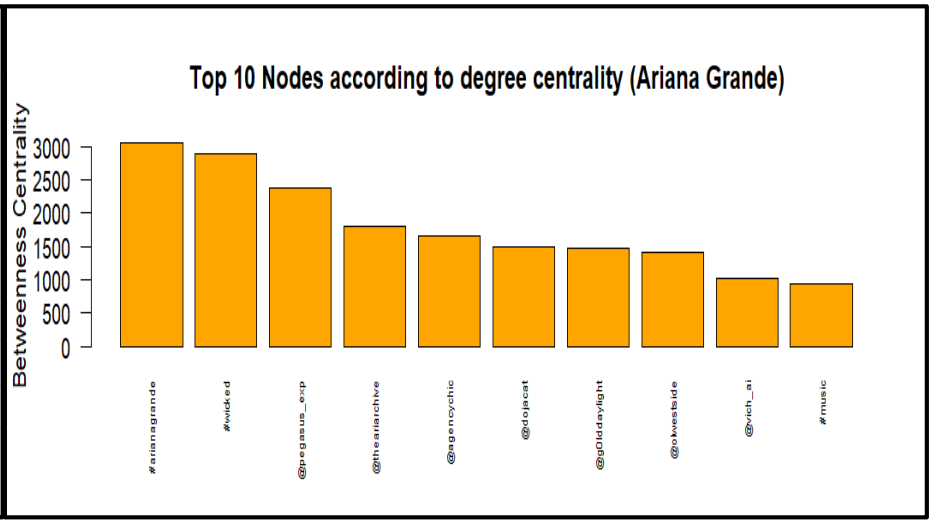
Now on the basis of closeness centrality:

	Closeness Centrality
@pegasus_exp	0.002433090
#wicked	0.002421308
#arianagrande	0.002421308
@agencychic	0.002132196
@vich_ai	0.002061856
@_bb_radio_music	0.002044990
@bbc2musicbot	0.002024291
#jonathanbailey	0.002020202
@ariscultt	0.002012072
@photoflix_ab	0.002012072



Finally on the basis of betweenness centrality:

	Betweenness Centrality
#arianagrande	3062
#wicked	2896
@pegasus_exp	2373
@theariarchive	1800
@agencychic	1664
@dojacat	1495
@g0lddaylight	1482
@olwestside	1422
@vich_ai	1031
#music	946



If you compare the three scores of both the artist you may come across the following observations

1) The nodes associated with the artist “The Weeknd” have higher degree centrality in comparison with Ariana Grande.
2) Similar observation can be inferred for closeness centrality
3) The betweenness centrality of the nodes of The Weeknd is also higher than that of Ariana Grande.

QUESTION 2.5

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?

Following piece of code is used to perform community analysis on artist The Weeknd:

```
########################################################################
########################  QUESTION NO 2.5  #############################
########################################################################

# Search YouTube
yt_oauth(app_id = client_id, app_secret = client_secret, token = '')


the_weeknd_video_search <- yt_search("The Weeknd")
ariana_grande_video_search <- yt_search("Ariana Grande")


the_weeknd_video_ids <- as.vector(the_weeknd_video_search$video_id[1:3])
ariana_grande_video_ids <- as.vector(ariana_grande_video_search$video_id[1:3])


the_weeknd_yt_data <- Authenticate("youtube", apiKey = api_key) %>%
  Collect(videoIDs = the_weeknd_video_ids,
          writeToFile = TRUE,
          maxComments = 250,
          verbose = TRUE)
ariana_grande_yt_data <- Authenticate("youtube", apiKey = api_key) %>%
  Collect(videoIDs = ariana_grande_video_ids,
          writeToFile = TRUE,
          maxComments = 250,
          verbose = TRUE)


View(the_weeknd_yt_data)
View(ariana_grande_yt_data)
```

Louvain Algorithm implementation

```
# Run Louvain algorithm
the_weeknd_louvain_yt_actor <- cluster_louvain(the_weeknd_undir_yt_actor_graph)
ariana_grande_louvain_yt_actor <- cluster_louvain(ariana_grande_undir_yt_actor_graph)
```

The Weeknd:

```
> sizes(the_weeknd_louvain_yt_actor)
Community sizes
  1   2   3   4   5   6   7   8   9  10
247 233   3 241   2   3   2   6   3   3
```

Ariana Grande:

```
> sizes(ariana_grande_louvain_yt_actor)
Community sizes
  1   2   3   4   5   6   7   8   9  10  11  12
211 201  10   4   9   2   2   3   3   3 224   3
```

Grirvan-Newman implementation

```
477  # Run Girvan-Newman (edge-betweenness) algorithm
478  the_weeknd_eb_yt_actor <- cluster_edge_betweenness(the_weeknd_undir_yt_actor_graph)
479  ariana_grande_eb_yt_actor <- cluster_edge_betweenness(ariana_grande_undir_yt_actor_graph)
480
```

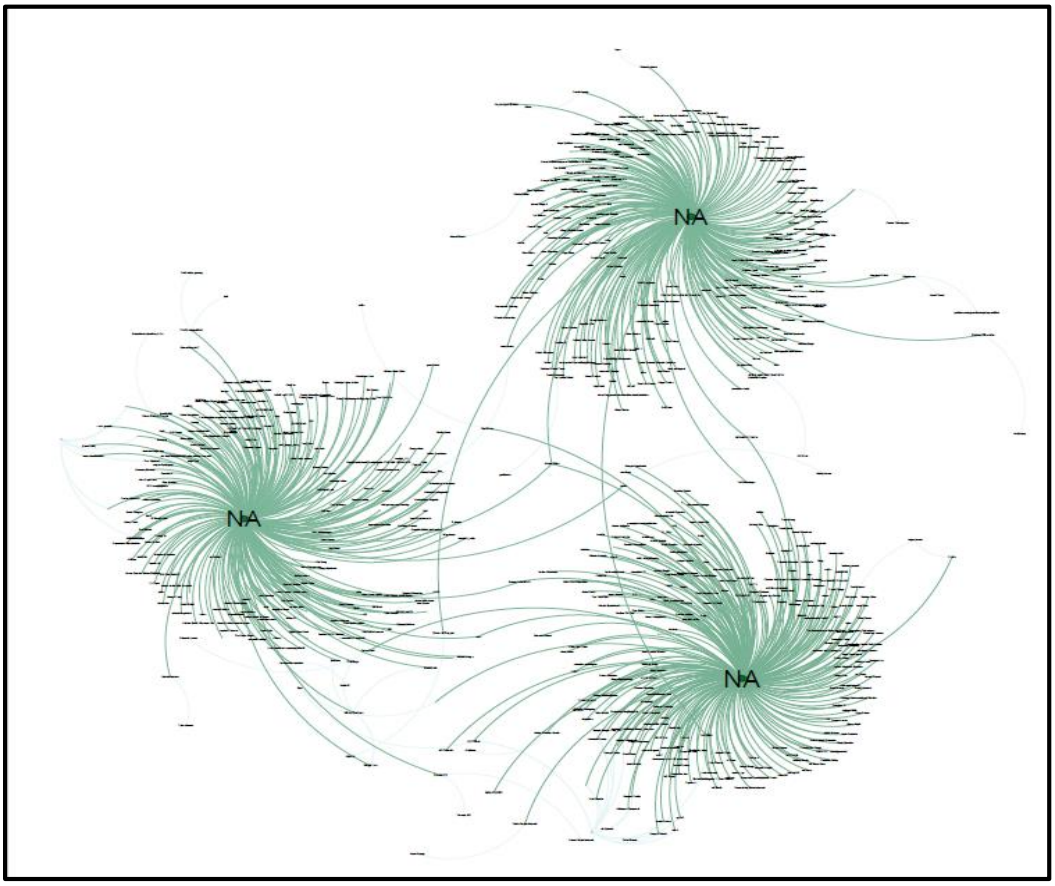
The Weeknd:

```
> sizes(the_weeknd_eb_yt_actor)
Community sizes
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19
240 235   2   2   2   2   2 233   2   2   2   2   2   2   3   2   2   3   3
```

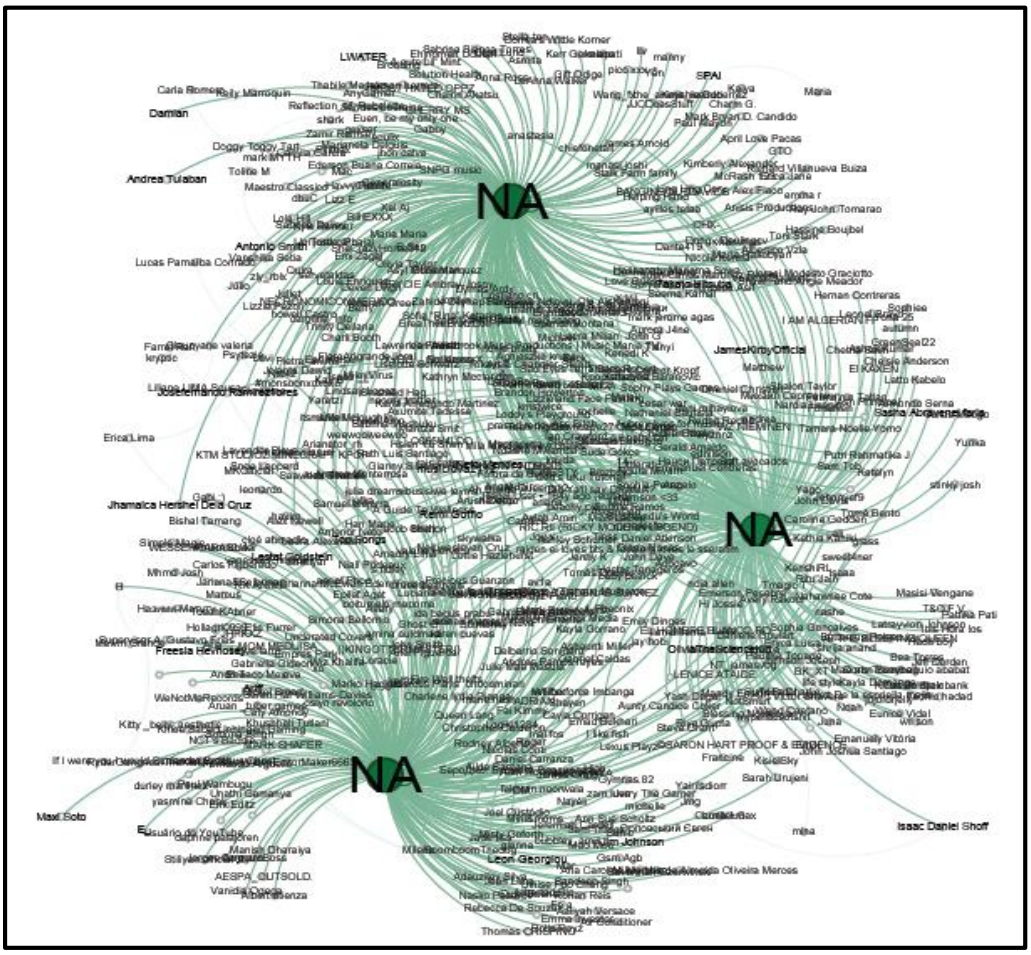
Ariana Grande:

```
> sizes(ariana_grande_eb_yt_actor)
Community sizes
  1   2   3   4   5   6   7   8   9
223  10 202   4   3   3   3 224   3
>
```

The following visualization depicts the implementation of Louvain Algorithm on The Weeknd:



The following visualization depicts the implementation of Louvain Algorithm on Ariana Grande:



Upon observing the graphs of both the artist we can see that most of the nodes form three dense communities.

QUESTION 2.6

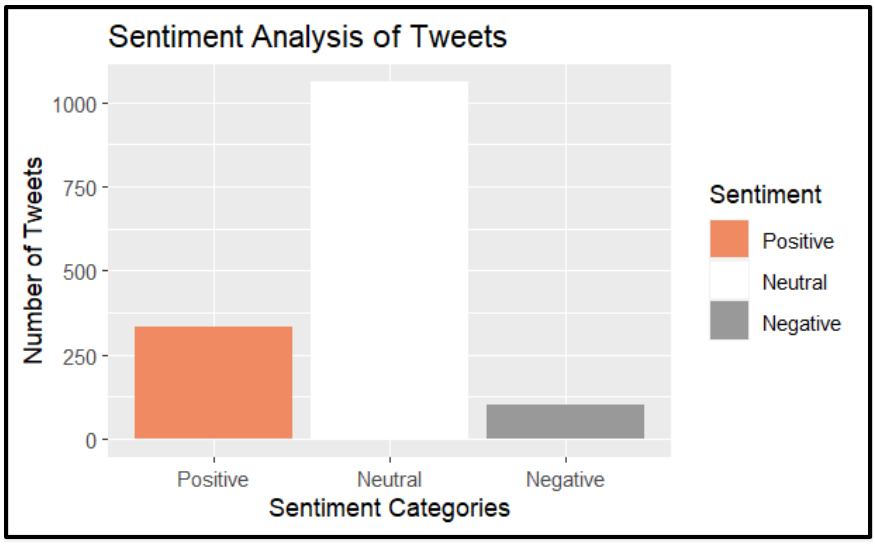
2.6) Use sentiment analysis to identify how the public reacts to events and/or topics related to your artist/band. Provide a summary of public opinions (emotions, reactions).

The following code is used to perform the sentiment analysis performed on the data that is Collected from twitter.

```
# Assign sentiment scores to tweets
the_weeknd_sentiment_scores <- get_sentiment(clean_text, method = "afinn") %>% sign()

the_weeknd_sentiment_df <- data.frame(text = clean_text, sentiment = the_weeknd_sentiment_scores)
View(the_weeknd_sentiment_df)


# Convert sentiment scores to labels: positive, neutral, negative

sentiment_df$sentiment <- factor(sentiment_df$sentiment, levels = c(1, 0, -1),
                                 labels = c("Positive", "Neutral", "Negative"))
View(sentiment_df)
```

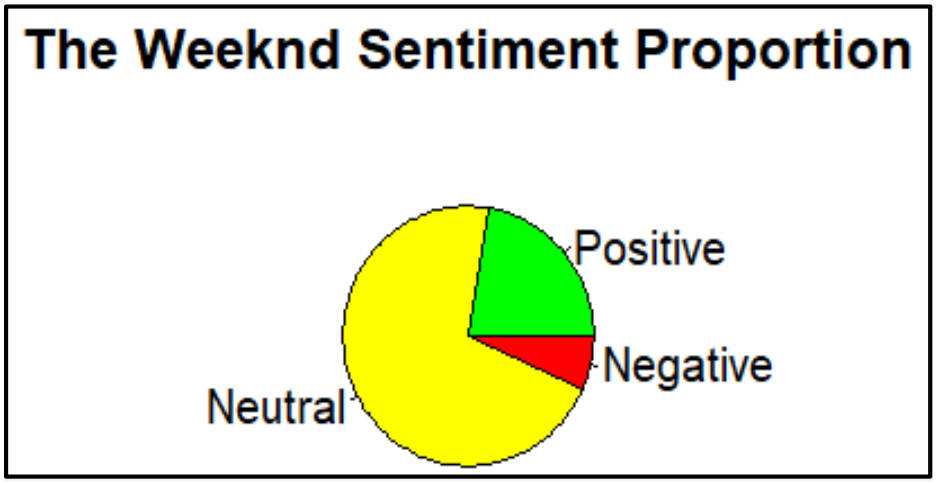
	text	sentiment
1	rt mementomorixo oh they coming with a banger for sure c...	Neutral
2	rt dariatysz i do not know if i am ready to say goodbye to th...	Neutral
3	rt dariatysz i do not know if i am ready to say goodbye to th...	Neutral
4	rt dariatysz does that mean we will get biopic about eyes	Neutral
5	rt yahoosg of teases potential new music with	Neutral
6	information s teases her collaboration with for hbo s the idol	Neutral
7	rt yahoosg of teases potential new music with	Neutral
8	rt dariatysz does that mean we will get biopic about eyes	Neutral
9	rt worldmusicawar tongue sticking out top arrow keycap so...	Positive
10	rt dariatysz i do not know if i am ready to say goodbye to th...	Neutral

The visualization correctly depicts how the data set is distributed on the basis of sentiments. From the graph it is clear that most of the tweets are neutral, followed by positive tweets and finally negative tweets.



Another way to comprehend the results is by plotting the distributions in a pie chart:

```
The_weeknd_sents <- data.frame(table(the_weeknd_sentiment_df['sentiment']))

slices <- The_weeknd_sents$Freq
lbls <- The_weeknd_sents$sentiment
colors = c("green", "yellow","red")
pie(slices, labels = lbls, main="The Weeknd Sentiment Proportion", col=colors)

```



Upon seeing the above two visualizations we can see that most of the public's sentiment is neutral followed by positive responses.

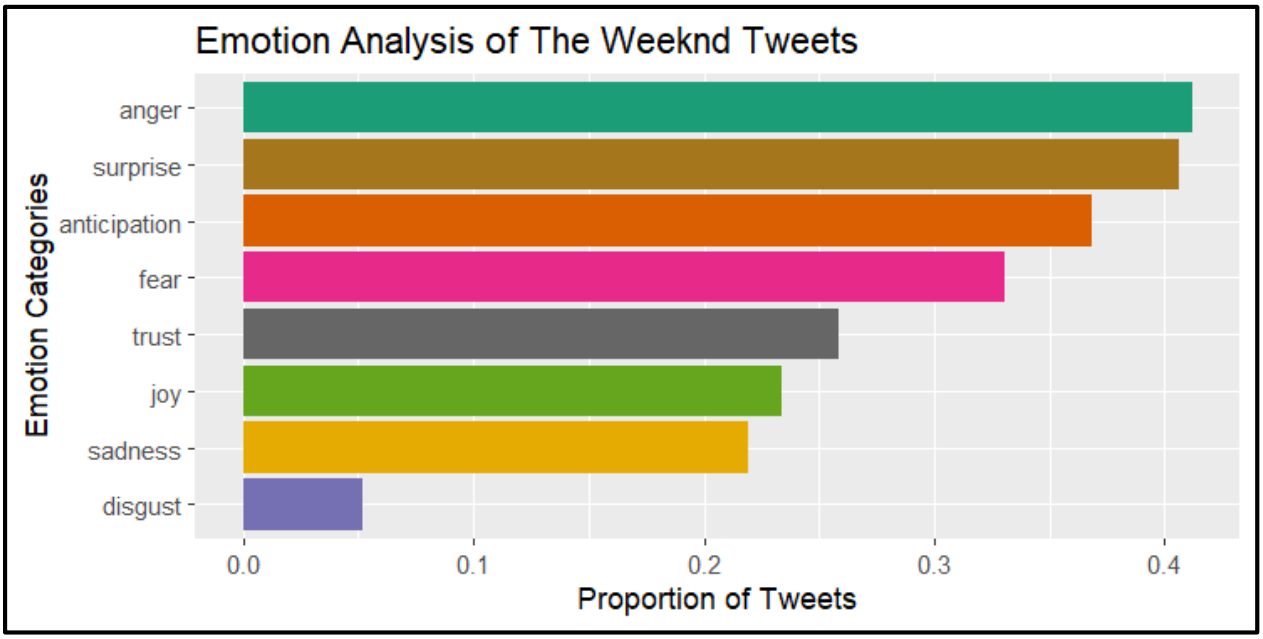
After that we calculate the emotions associated to the tweets.

```
# Calculate proportion of emotions across all tweets
the_weeknd_emo_sums <- the_weeknd_emo_scores_df[,2:9] %>%
  sign() %>%
  colSums() %>%
  sort(decreasing = TRUE) %>%
  data.frame() / nrow(emo_scores_df)

names(the_weeknd_emo_sums)[1] <- "Proportion"
View(the_weeknd_emo_sums)
```

	clean_text	anger	anticipation	disgust	fear	joy	sadness	surprise	trust
1	rt mementomorixo oh they coming with a banger for sure c...	2	2	0	2	0	1	1	0
2	rt dariatysz i do not know if i am ready to say goodbye to th...	0	1	0	0	0	0	0	1
3	rt dariatysz i do not know if i am ready to say goodbye to th...	0	1	0	0	0	0	0	1
4	rt dariatysz does that mean we will get biopic about eyes	0	0	0	0	0	0	0	0
5	rt yahoosg of teases potential new music with	0	0	0	0	1	1	0	0
6	information s teases her collaboration with for hbo s the idol	0	0	0	0	0	0	0	0
7	rt yahoosg of teases potential new music with	0	0	0	0	1	1	0	0
8	rt dariatysz does that mean we will get biopic about eyes	0	0	0	0	0	0	0	0
9	rt worldmusicawar tongue sticking out top arrow keycap so...	1	2	0	0	3	2	1	3
10	rt dariatysz i do not know if i am ready to say goodbye to th...	0	1	0	0	0	0	0	1

Finally we plot the results

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



However if refer to the emotion analysis conducted for The Weeknd, the model has predicted anger and surprise as the most frequent categories.

QUESTION 2.7

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

This is the data set that we are going to use to build a decision tree.

	artist_name	artist_id	album_id	album_type	album_images	album_release_date	album_release_year
1	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
2	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
3	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
4	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
5	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
6	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
7	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
8	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
9	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
10	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023
11	The Weeknd	1Xyo4u8uXC1ZmMpatF05PJ	35dut3ICqF3NEDkjxfzJJ1	album	*2 variables*	2023-03-14	2023

To We split the data set into train and test sets, with a 80/20 approach, 80% of dataset kept for training and 20% of data for testing the accuracy of the model.

```
643 # Split the dataset into training and testing sets (80% training, 20% testing)
644 split_point <- as.integer(nrow(comb_data)*0.8)
645 training_set <- comb_data[1:split_point, ]
646 testing_set <- comb_data[(split_point + 1):nrow(comb_data), ]
647
648
649 # Train the decision tree model
650 dt_model <- train(isTheWeeknd~ ., data = training_set, method = "C5.0")
651
```

```
664 # Analyse the model accuracy with a confusion matrix
665 confusionMatrix(dt_model, reference = testing_set$isTheWeeknd)
666
```

After that we check the performance of the decision tree using the confusion matrix.

```
          Reference
Prediction    0    1
         0  6.5  1.4
         1  6.0 86.1

 Accuracy (average) : 0.9263
```

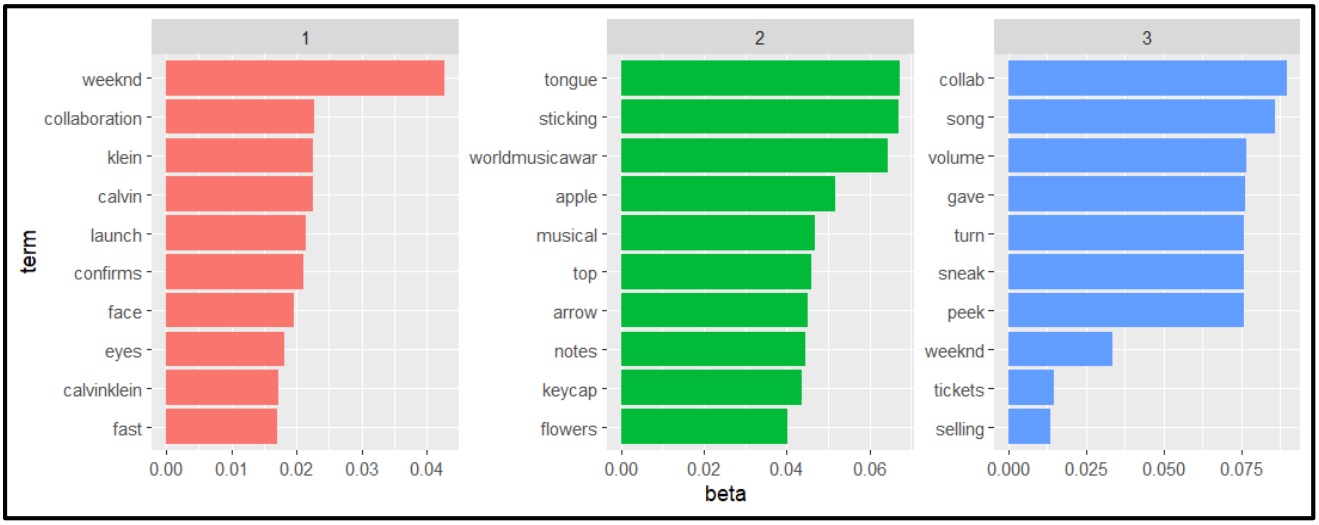
As per the above results the accuracy of the model comes to 92.63%

QUESTION 2.8

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.

We have used the following code to perform LDA on our dataset and selected 3 groups our result.

```
top_terms <- tweet_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()
```



The first group talks about the collaboration that has just happened between The Weeknd and Calvin Klein.

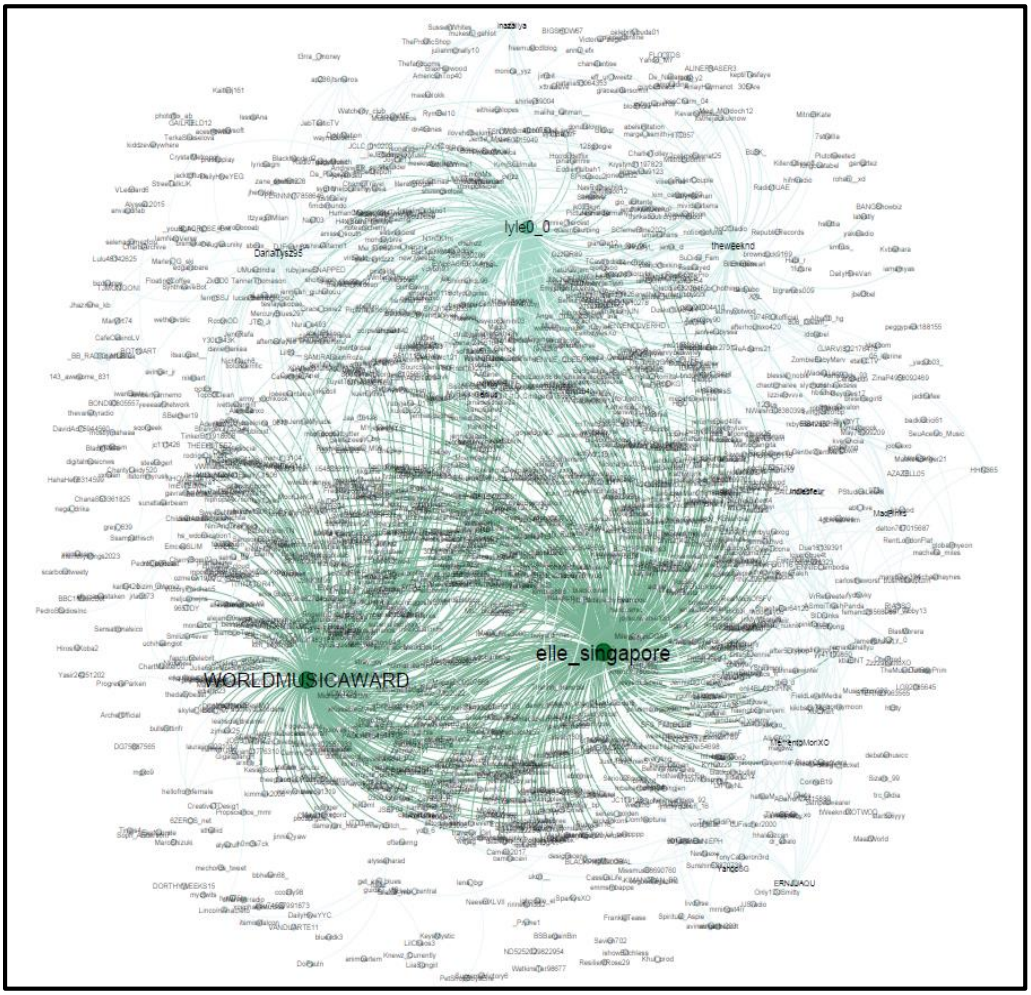
The second group of words are associated with world music award which is an international organization that recognizes the popular singers.

The last group talks something regarding the collaboration of the Weeknd however a crucial aspect which is missing from the group of words is missing that is who the other collaborator was.

QUESTION 2.9

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.)

The following graph depicts the twitter actor network and the size of a node denotes how many follower each node has.



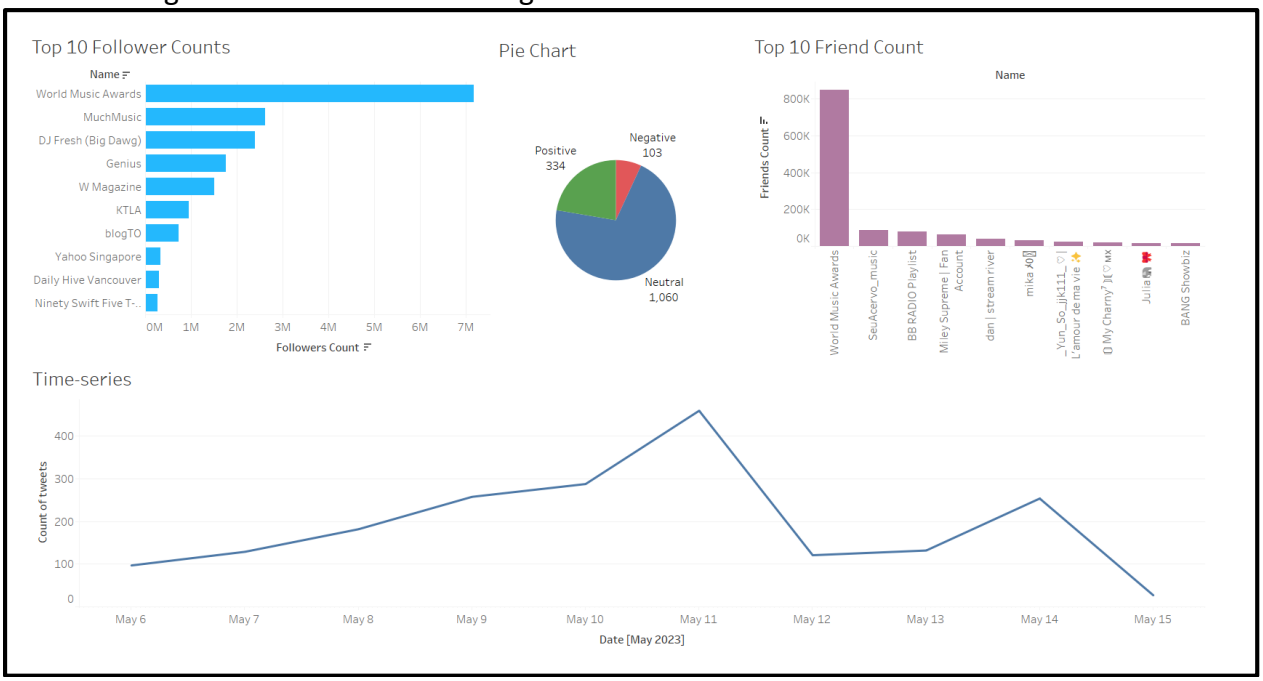
From the graph above you can see the three dominant nodes are:

- World Music Awards
- Elle_singapore
- Lyle0_0

QUESTION 2.10

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.

The following dashboard is created using the tableau:



Visualization	Interpretation
Horizontal Bar chart	This helps in identifying the top 10 users on the basis of follower count. This would help in marketing campaigns as these individuals have influence over large number of individuals and would help in reaching large audience.
Pie Chart	Depicts the proportion of positive, negative and neutral sentiments, helps in visualizing how we are performing among the masses.
Vertical Bar chart	Lists down top 10 individuals on the basis of friends they have, the reason is same as the first visualization that is to reach larger audience using influential nodes.
Time series Analysis	Visualizes how many tweets were posted on a particular day, helps to see when was the trend, or the artist was talk of the town.

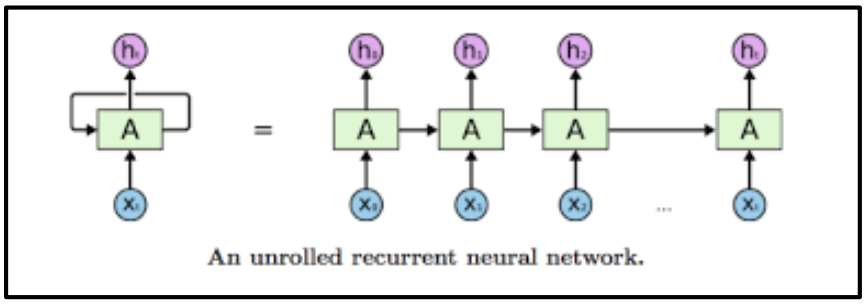
QUESTION 2.11

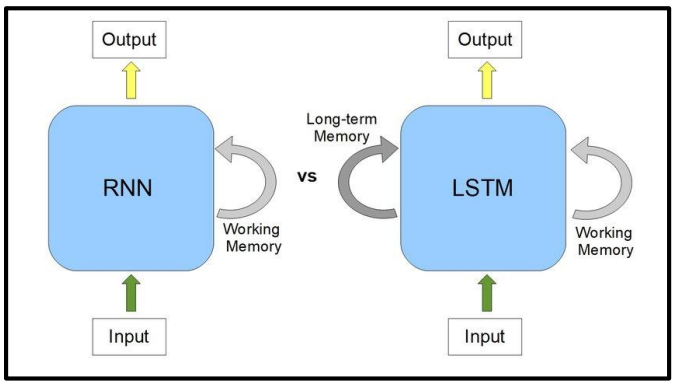
2.11) Research and review other methods/algorithms for network analysis, machine learning models, or visualisation. Compare them to the methods you used in these milestones. Did you find a method that could give you better insights or more promising results for your social media analytics? Explain why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

MACHINE LEARNING:

The machine learning model that was able to find for the purpose of sentiment prediction in this particular case comes under the domain of deep learning called RNN or recurrent neural network.

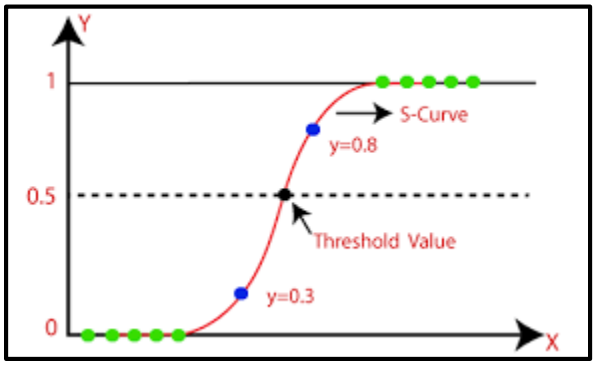
That is not all, RNN with LSTM. The way RNN works is it forwards the context of the previous node in order to predict the sentiment of the given dataset. But the problem with this model is that for large chains the context loses the weight and might not carry the same weight till the last node, this is where the LSTM plays a vital role. LSTM stand for Long Short Term Memory and helps in retaining the context throughout the network.





By opting for deep learning approach, there is considerable evidence to conclude that it would yield better results in the prediction of sentiments.

P. Sujan Reddy and et al. used logistic regression with grid search which gave an accuracy of 94% . Alongside the model proposed they also highlighted the importance of the polarity which ultimately leads to the prediction of sentiment



So from the techniques reviewed following two models can also be used for the prediction of sentiments:

1) RNN + LSTM (deep learning approach)
2) Support Vector Machine (Supervised ML techniques)

VISUALIZATION:

Data extraction and representation methods are highly desirable to assist user to comprehend the huge data.

Research conducted by Milka Trajkova and et al. reports qualitative and quantitative findings from a Twitter crawl of 5409 posts during the period of the COVID-19 epidemic and to present these findings they have used the line graphs, bar charts, pie charts, scatter plots and stream line visualization techniques.

A fascinating approach was explored by visualizing the sentiments of the US presidential elections and UK general elections and plotting them state wise by Ussama Yaqub and et. al. This concept can be used to explore the idea of how a particular region in the world views the trending topic

Upon reviewing the aforementioned papers one can suggest the following techniques that can be used for visualization:

1) Line Chart (for time series analysis)
2) Scatter Plot (for correlation analysis)
3) Geographical heat map

REFERENCES:

[TensorFlow]. (2020, March 4). *Natural Language Processing (NLP) Zero to Hero* [Video]. YouTube. https://www.youtube.com/playlist?list=PLQY2H8rRoyvzDbLUZkbudP-MFQZwNmU4S

Reddy, P. S., Sri, D. R., Reddy, C. S., & Shaik, S., Dr. (2020). Sentimental Analysis using Logistic Regression. *International Journal of Engineering Research and Applications*, *11*(7), 36-40. https://doi.org/10.9790/9622-1107023640

Trajkova, M., Alhakamy, A., Cafaro, F., Vedak, S., Mallappa, R., & Kankara, S. R. (2020). Exploring Casual COVID-19 Data Visualizations on Twitter: Topics and Challenges. *Informatics*, *7*(3), 35. https://doi.org/10.3390/informatics7030035

YAQUB, U., SHARMA, N., & PABREJA, R. (2020). Location-based Sentiment Analyses and Visualization of Twitter Election Data. *Digital Government: Research and Practice*, *1*(2). https://doi.org/10.1145/3339909