|  |
| --- |
|  |
| Group Assignment: Web & Social Media Analytics |
| Group 8 |
|  |
| **Jyotirmoy Dutta**  **Rudra Chakraborty**  **Neeraj Gupta**  **Dr. V.L.Narshiman** |
| **7/21/2019** |

Table of Contents

[I. Brand and Extracting tweets using R Functions : 2](#_Toc14528493)

[II. Creating Dataframe with Tweet data: 4](#_Toc14528494)

[III. Cleaning ,Transforming and Normalizing data for analysis 6](#_Toc14528495)

[IV. Frequency Analysis and Visualizations : 7](#_Toc14528496)

[V. Term association and Sentiment Analysis 13](#_Toc14528497)

[VI. Conclusion: 20](#_Toc14528499)

[VII. Appendix: 20](#_Toc14528500)

# Brand and Extracting tweets using R Functions:

Premium streaming platform, **Hotstar** has set a new global benchmark during the India vs New Zealand semi-final on 10th July 2019. The platform registered an unprecedented 25.3 million simultaneous viewership. Hotstar was Hot (topic) during the Cricket World cup and Wimbledon session. But what was the overall level of experience people had during this time? To get an idea we chose Twitter as platform to collect Twitteraties' reactions on Hotstar. We collected data from July 6 to July 14, 2019, the happening 9 days of world of sports, when we watched live the Greatest CWC final ever and caught the spectacular, longest-ever Wimbledon final at the same time.

We will use R as tool for twitter data analysis. First, we will install all the required packages:

install.packages("bit64")

install.packages("twitteR")

install.packages("ROAuth")

install.packages("SnowballC")

install.packages("tm")

install.packages("ggplot2")

install.packages("RColorBrewer")

install.packages("wordcloud")

install.packages("topicmodels")

install.packages("data.table")

install.packages("stringi")

install.packages("syuzhet")

install.packages("dplyr")

install.packages("plyr")

install.packages("grid")

install.packages("gridExtra")

====================================================================

Loading the above packages

library(bit64)

library(twitteR)

library(ROAuth)

library(SnowballC)

library(tm)

library(ggplot2)

library(RColorBrewer)

library(wordcloud)

library(topicmodels)

library(data.table)

library(stringi)

library(syuzhet)

library(dplyr)

library(plyr)

library(grid)

library(gridExtra)

=================================================================================

After loading all the packages, we will invoke Twitter API, using the app we have created and using the keys and access tokens we got through the app.

> api\_key <- "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"

> api\_secret <- "XXXXXXXXXXXXXXXXXXXXXXXXXXXXX"

> access\_token <- "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"

> access\_token\_secret <- "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"

> setup\_twitter\_oauth(api\_key,api\_secret,access\_token,access\_token\_secret)

For string "#Hotstar" we will try to fetch around 2000 tweets

> search.string <- "#Hotstar"

> no.of.tweets <- 2000

> HS\_tweets <- searchTwitter(search.string, n=no.of.tweets,lang="en")

Warning message:

In doRppAPICall("search/tweets", n, params = params, retryOnRateLimit = retryOnRateLimit, :

2000 tweets were requested but the API can only return 1100

The API we used could only fetch 1108 tweets which is ok for our group assignment problem.

> n.tweet <- length(HS\_tweets)

> n.tweet

[1] 1100

# Creating Dataframe with Tweet data:

We have to put the extracted data into a dataframe and check the dataframe format.

#Creating dataframe with extracted tweets

> HS\_tweets\_df <- twListToDF(HS\_tweets)

#Dimension of the data frame

> dim(HS\_tweets\_df)

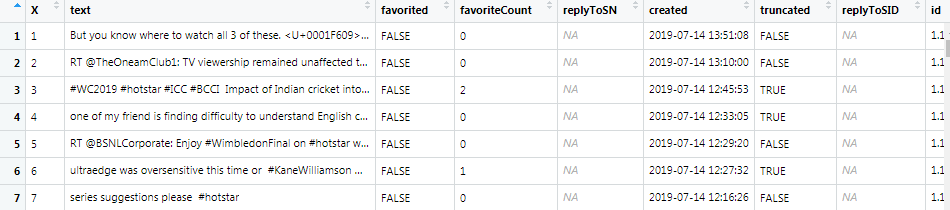
[1] 1100 16

The dataframe has 16 variables with 1100 observations.

#View the dataframe

> View(HS\_tweets\_df)

Snapshot of the dataframe.



Writing the dataframe into csv

> setwd("F:\\Web & Social Media Analytics\\GA")

> write.csv(HS\_tweets\_df,file=paste("HS\_tweets\_df.csv"))



Reading the dataset into R in desired format

tweets.df<-read.csv("HS\_tweets\_df.csv",stringsAsFactors = FALSE)

This is how current text fields of the data frame looks like

> HS\_tweets <- tweets.df$text

> head(HS\_tweets)

[1] "But you know where to watch all 3 of these. <U+0001F609> #hotstar https://t.co/FPR8in0D5c"

[2] "RT @TheOneamClub1: TV viewership remained unaffected this June, in spite of the fact that many people took to streaming service #Hotstar to…"

[3] "#WC2019 #hotstar #ICC #BCCI \n\nImpact of Indian cricket into the world cricket it can be explicitly seen, in every m… https://t.co/QcnXChYr1q"

[4] "one of my friend is finding difficulty to understand English commentary,i am sure there are many like him. Is it po… https://t.co/oHdP7choSk"

[5] "RT @BSNLCorporate: Enjoy #WimbledonFinal on #hotstar with #BSNL Superstar #300GB Pack. To activate please call on 1800-345-1500. \n#Wimbledo…"

[6] "ultraedge was oversensitive this time or \n#KaneWilliamson was underspirited bcz its #CWC19Final ¿ \n\n@BLACKCAPS… https://t.co/iAWteEFzYa"

If we see the structure of the dataframe the created column is in CHAR format.

|  |
| --- |
| > str(tweets.df)  'data.frame': 1100 obs. of 17 variables:  $ X : int 1 2 3 4 5 6 7 8 9 10 ...  $ text : chr "But you know where to watch all 3 of these. <U+0001F609> #hotstar https://t.co/FPR8in0D5c" "RT @TheOneamClub1: TV viewership remained unaffected this June, in spite of the fact that many people took to s"| \_\_truncated\_\_ "#WC2019 #hotstar #ICC #BCCI \n\nImpact of Indian cricket into the world cricket it can be explicitly seen, in e"| \_\_truncated\_\_ "one of my friend is finding difficulty to understand English commentary,i am sure there are many like him. Is i"| \_\_truncated\_\_ ...  $ favorited : logi FALSE FALSE FALSE FALSE FALSE FALSE ...  $ favoriteCount: int 0 0 2 0 0 1 0 0 0 0 ...  $ replyToSN : chr NA NA NA NA ...  $ created : chr "2019-07-14 13:51:08" "2019-07-14 13:10:00" "2019-07-14 12:45:53" "2019-07-14 12:33:05" ...  $ truncated : logi FALSE FALSE TRUE TRUE FALSE TRUE ...  $ replyToSID : num NA NA NA NA NA NA NA NA NA NA ...  $ id : num 1.15e+18 1.15e+18 1.15e+18 1.15e+18 1.15e+18 ...  $ replyToUID : num NA NA NA NA NA NA NA NA NA NA ...  $ statusSource : chr "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for iPhone</a>" "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" "<a href=\"https://mobile.twitter.com\" rel=\"nofollow\">Twitter Web App</a>" "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" ...  $ screenName : chr "thejas\_babu" "rahulrrmodi" "piyushg73946435" "nishitdpatel" ...  $ retweetCount : int 0 1 0 0 9 0 0 4 9 0 ...  $ isRetweet : logi FALSE TRUE FALSE FALSE TRUE FALSE ...  $ retweeted : logi FALSE FALSE FALSE FALSE FALSE FALSE ...  $ longitude : logi NA NA NA NA NA ...  $ latitude : logi NA NA NA NA NA NA ... |
|  |
| |  | | --- | |  | |

We need to convert “created” column into Date format.

> tweets.df$created <- as.Date(tweets.df$created)

> str(tweets.df)

'data.frame': 1100 obs. of 17 variables:

$ X : int 1 2 3 4 5 6 7 8 9 10...

$ text : chr "But you know where to watch all 3 of these. <U+0001F609> #hotstar https://t.co/FPR8in0D5c" "RT @TheOneamClub1: TV viewership remained unaffected this June, in spite of the fact that many people took to s"| \_\_truncated\_\_ "#WC2019 #hotstar #ICC #BCCI \n\nImpact of Indian cricket into the world cricket it can be explicitly seen, in e"| \_\_truncated\_\_ "one of my friend is finding difficulty to understand English commentary,i am sure there are many like him. Is i"| \_\_truncated\_\_ ...

$ favorited : logi FALSE FALSE FALSE FALSE FALSE FALSE ...

$ favoriteCount: int 0 0 2 0 0 1 0 0 0 0 ...

$ replyToSN : chr NA NA NA NA ...

$ created : Date, format: "2019-07-14" "2019-07-14" "2019-07-14" "2019-07-14" ...

$ truncated : logi FALSE FALSE TRUE TRUE FALSE TRUE ...

$ replyToSID : num NA NA NA NA NA NA NA NA NA NA ...

$ id : num 1.15e+18 1.15e+18 1.15e+18 1.15e+18 1.15e+18 ...

$ replyToUID : num NA NA NA NA NA NA NA NA NA NA ...

$ statusSource : chr "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for iPhone</a>" "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" "<a href=\"https://mobile.twitter.com\" rel=\"nofollow\">Twitter Web App</a>" "<a href=\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" ...

$ screenName : chr "thejas\_babu" "rahulrrmodi" "piyushg73946435" "nishitdpatel" ...

$ retweetCount : int 0 1 0 0 9 0 0 4 9 0 ...

$ isRetweet : logi FALSE TRUE FALSE FALSE TRUE FALSE ...

$ retweeted : logi FALSE FALSE FALSE FALSE FALSE FALSE ...

$ longitude : logi NA NA NA NA NA NA ...

$ latitude : logi NA NA NA NA NA NA ...

# Cleaning, Transforming and Normalizing data for analysis

This step involves transforming the text uniformly.

* Creating document corpus with Tweet text for further data transformation/cleaning:

> myCorpus<- Corpus(VectorSource(tweets.df$text))

* Convert Corpus to lowercase

> myCorpus <- tm\_map(myCorpus, content\_transformer(stri\_trans\_tolower))

> writeLines(strwrap(myCorpus[[700]]$content,60))

@hotstartweets @hotstar\_helps #hotstar you get most

subscription only because of ipl and world cup if this is

how y… https://t.co/W67uWfmHOU

* Remove the links (URLs)

> removeURL <- function(x) gsub("http[^[:space:]]\*", "", x)

> myCorpus <- tm\_map(myCorpus, content\_transformer(removeURL))

> writeLines(strwrap(myCorpus[[700]]$content,60))

@hotstartweets @hotstar\_helps #hotstar you get most

subscription only because of ipl and world cup if this is

how y…

* Remove the @ (usernames)

> removeUsername <- function(x) gsub("@[^[:space:]]\*", "", x)

> myCorpus <- tm\_map(myCorpus, content\_transformer(removeUsername))

> writeLines(strwrap(myCorpus[[700]]$content,60))

#hotstar you get most subscription only because of ipl and

world cup if this is how y…

* Remove anything except the English language and space

> removeNumPunct <- function(x) gsub("[^[:alpha:][:space:]]\*", "", x)

> myCorpus <- tm\_map(myCorpus, content\_transformer(removeNumPunct))

> writeLines(strwrap(myCorpus[[700]]$content,60))

hotstar you get most subscription only because of ipl and

world cup if this is how y

* Remove stop words. We have also identified our own stopwords and removed them.

> myStopWords<- c((stopwords('english')),c("rt", "use", "used", "via", "amp","user","hotstar","hotst","uf","da"))

> myCorpus<- tm\_map(myCorpus,removeWords , myStopWords)

Warning message:

> writeLines(strwrap(myCorpus[[700]]$content))

get subscription ipl world cup y

|  |
| --- |
| * Remove Single letter words   > removeSingle <- function(x) gsub(" . ", " ", x)  > myCorpus <- tm\_map(myCorpus, content\_transformer(removeSingle))  Warning message:  In tm\_map.SimpleCorpus(myCorpus, content\_transformer(removeSingle)) :  transformation drops documents  > writeLines(strwrap(myCorpus[[1100]]))  RT still heart broken KahaanHumKahaanTum UF Very hug disappointed TRP hope future UF good job  Starplus |
|  |
| |  | | --- | |  | |
| * Remove extra whitespace   > myCorpus<- tm\_map(myCorpus, stripWhitespace)  Warning message:  In tm\_map.SimpleCorpus(myCorpus, stripWhitespace) :  transformation drops documents  > writeLines(strwrap(myCorpus[[804]]))  HotStar sucks Is just office guys suffering technical failure NZvIND CWC   * Stemming :   > myCorpus<- tm\_map(myCorpus, stemDocument)  > writeLines(strwrap(myCorpus[[221]]))  kill technic failur interrupt cwc nzvind |

# Frequency Analysis and Visualizations :

|  |
| --- |
| * Keeping a copy of "myCorpus" for stem completion later     > myCorpusCopy<- myCorpus  **Term Document Matrix**: The term-document matrix then is a two-dimensional matrix whose rows are the  terms and columns are the documents.  **tm** package provides a function called **TermDocumentMatrix** that generates the matrix  By creating Term document Matrix our next step is to find the most frequent words/terms in the tweets.   * Creating a term document matrix:   > tdm<- TermDocumentMatrix(myCorpus)  > tdm  Now we have the word frequencies used in our twitter data.  The values in the matrix are the counts of how many times that word appeared in each document.  <<TermDocumentMatrix (terms: 1693, documents: 1100)>>  Non-/sparse entries: 8356/1853944  Sparsity : 100%  Maximal term length: 29  Weighting : term frequency (tf)   * Find the terms used most frequently. Here we are listing down the terms which has more than 25 occurrences.   > (freq.terms <- findFreqTerms(tdm, lowfreq = 25))  [1] "watch" "streaming" "viewership" "world" "activate"  [6] "bsnl" "call" "enjoy" "pack" "please"  [11] "superstar" "experience" "now" "tournament" "wimbledon"  [16] "india" "match" "watching" "million" "record"  [21] "worldcup" "netflix" "new" "today" "live"  [26] "semifinal" "app" "highest" "subscription" "will"  [31] "indiavsnewzealand" "indvnz" "indvsnz" "cwc" "second"  [36] "failure" "interruptions" "killing" "nzvind" "technical"  > term.freq <- rowSums(as.matrix(tdm))  > term.freq <- subset(term.freq, term.freq > 25)   * Let's put the Frequency Distribution table into dataframe **df**   > df <- data.frame(term = names(term.freq), freq= term.freq)  > df  term freq  watch watch 81  streaming streaming 54  viewership viewership 27  world world 54  activate activate 203  bsnl bsnl 203  call call 203  enjoy enjoy 93  pack pack 204  please please 219  superstar superstar 204  experience experience 75  now now 115  tournament tournament 157  wimbledon wimbledon 157  india india 57  match match 48  watching watching 28  million million 45  record record 26  worldcup worldcup 63  netflix netflix 33  new new 36  today today 36  live live 65  semifinal semifinal 26  app app 28  highest highest 28  subscription subscription 27  will will 27  indiavsnewzealand indiavsnewzealand 37  indvnz indvnz 64  indvsnz indvsnz 57  cwc cwc 312  second second 29  failure failure 253  interruptions interruptions 250  killing killing 248  nzvind nzvind 250  technical technical 271   * Here we are listing down the terms which has more than 10 occurrences.     > (freq.terms <- findFreqTerms(tdm, lowfreq = 10))  [1] "watch" "people" "streaming" "viewership" "can"  [6] "cricket" "icc" "world" "like" "one"  [11] "sure" "activate" "bsnl" "call" "enjoy"  [16] "pack" "please" "superstar" "time" "experience"  [21] "now" "tournament" "wimbledon" "india" "match"  [26] "watching" "final" "just" "million" "sports"  [31] "viewers" "dhoni" "best" "going" "akamai"  [36] "record" "worldcup" "download" "team" "netflix"  [41] "dont" "new" "prime" "cup" "show"  [46] "day" "today" "hero" "hours" "woop"  [51] "play" "live" "concurrent" "semifinal" "app"  [56] "even" "whatsapp" "ever" "highest" "zealand"  [61] "money" "subscription" "get" "will" "indiavsnewzealand"  [66] "indvnz" "come" "users" "indvsnz" "cwc"  [71] "biggbosstamil" "event" "hits" "kind" "none"  [76] "othe" "second" "semi" "theofficeindia" "iccworldcup"  [81] "starsports" "failure" "interruptions" "killing" "nzvind"  [86] "technical" "whats" "paid" "annual" "cant"  [91] "still" "rain" "coz" "heartbreak" "meanwhile"  [96] "shituf" "indvssl" "announcement" "budget" "declared"  [101] "subsidy" "wim"    > term.freq <- rowSums(as.matrix(tdm))  > term.freq <- subset(term.freq, term.freq > 10)   * Let's put the Frequency Distribution table into dataframe **df1**   > df1 <- data.frame(term = names(term.freq), freq= term.freq)   * Same way we put frequency of more than 55 and 85 occurrences of terms in **df2** and **df3** dataframes.     > (freq.terms <- findFreqTerms(tdm, lowfreq = 55))  [1] "watch" "activate" "bsnl" "call" "enjoy" "pack"  [7] "please" "superstar" "experience" "now" "tournament" "wimbledon"  [13] "india" "worldcup" "live" "indvnz" "indvsnz" "cwc"  [19] "failure" "interruptions" "killing" "nzvind" "technical"  > term.freq <- rowSums(as.matrix(tdm))  > term.freq <- subset(term.freq, term.freq > 55)  > df2 <- data.frame(term = names(term.freq), freq= term.freq)  > (freq.terms <- findFreqTerms(tdm, lowfreq = 85))  [1] "activate" "bsnl" "call" "enjoy" "pack" "please"  [7] "superstar" "now" "tournament" "wimbledon" "cwc" "failure"  [13] "interruptions" "killing" "nzvind" "technical"  > term.freq <- rowSums(as.matrix(tdm))  > term.freq <- subset(term.freq, term.freq > 85)  > df3 <- data.frame(term = names(term.freq), freq= term.freq)   * Plotting the graph of frequent terms   > p1=ggplot(df1, aes(reorder(term, freq),freq)) + theme\_bw() + geom\_bar(stat = "identity") + coord\_flip() +labs(list(title="@10", x="Terms", y="Term Counts")) + theme(axis.text.y = element\_text(size=7))    > p2=ggplot(df, aes(reorder(term, freq),freq)) + theme\_bw() + geom\_bar(stat = "identity") + coord\_flip() +labs(list(title="@25", x="Terms", y="Term Counts"))+  + theme(axis.text.y = element\_text(size=7))    > p3=ggplot(df2, aes(reorder(term, freq),freq)) + theme\_bw() + geom\_bar(stat = "identity") + coord\_flip() +labs(list(title="@55", x="Terms", y="Term Counts"))  > p4=ggplot(df3, aes(reorder(term, freq),freq)) + theme\_bw() + geom\_bar(stat = "identity") + coord\_flip() +labs(list(title="@85", x="Terms", y="Term Counts"))   * Combining the plots   > grid.arrange(p1,p2,ncol=2)  **Frequency Barplots of more than 10 and 25 terms:**  Rplot02.png  > grid.arrange(p3,p4,ncol=2)  Frequency Barplots of more than 55 and 85 terms:  Rplot03.png |
|  |
| |  | | --- | |  | |

**Inference from the frequency Barplot:**

1. Twitteraties' most discussed topic related to Hotstar was CWC i.e. ICC Cricket Worldcup.
2. India vs. Newzeland nail biting semifinal match was one of the hot topics. Hotstar set a new

record of highest viewership on that day – 25.1 M on July10th 2019.

1. We can see there are also few negative terms appear in the Barplot. eg. Faliure, Killing,

interruptions etc. These show people brought out their frustration through twitter.

* Calculate the frequency of words and sort it by frequency and setting up the Wordcloud

> set.seed(777)

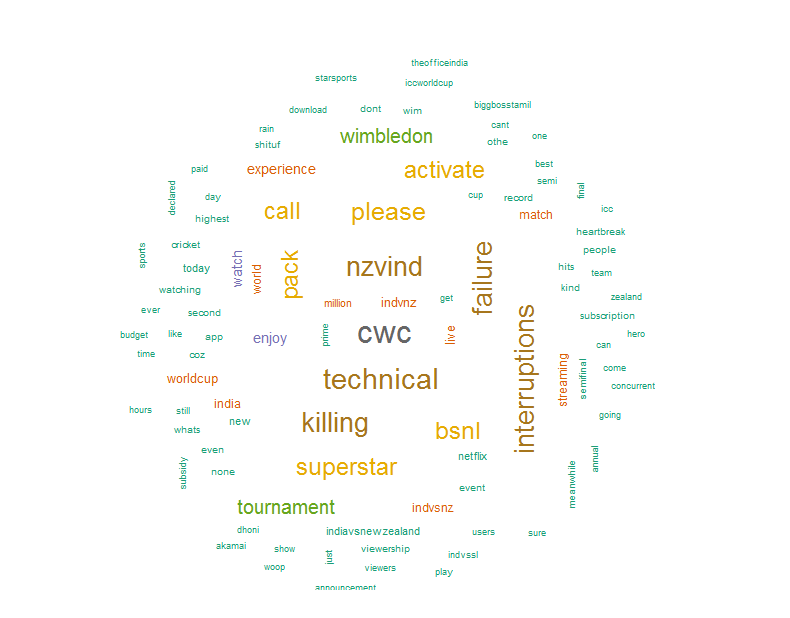
> word.freq <-sort(rowSums(as.matrix(tdm)), decreasing= TRUE)

> word.freq

> pal<- brewer.pal(8, "Dark2")

* **Visualizing tweets as a word cloud to find out what people are tweeting about Hotstar**

> wordcloud(words = names(word.freq), freq = word.freq, min.freq = 10,colors = pal,use.r.layout=T, scale=c(2,0.5),max.words =100, random.order=F)



The order of words is completely random but the size of the words is directly proportional to the

frequency of occurrence of the word in text files.

# Term association and Sentiment Analysis

Finding association with specific keyword in the tweets -

Association of the term "failure" with other terms

> list1<- findAssocs(tdm, "failure", 0.5)

> corrdf1 <- t(data.frame(t(sapply(list1,c))))

> corrdf1

failure

killing 0.98

interruptions 0.97

nzvind 0.97

technical 0.95

cwc 0.82

Association of the term "nzvind" with other terms (India vs. Newzeland match)

> list2<- findAssocs(tdm, "nzvind", 0.5)

> corrdf2 <- t(data.frame(t(sapply(list2,c))))

> corrdf2

nzvind

killing 0.98

failure 0.97

interruptions 0.97

technical 0.92

cwc 0.82

Association of the term "wimbledon" with other terms

> list3<- findAssocs(tdm, "wimbledon", 0.5)

> corrdf3 <- t(data.frame(t(sapply(list3,c))))

> corrdf3

wimbledon

tournament 0.99

activate 0.85

bsnl 0.85

call 0.85

pack 0.85

superstar 0.85

please 0.81

experience 0.65

enjoy 0.64

Association of the term "activate" with other terms

> list4<- findAssocs(tdm, "activate", 0.5)

> corrdf4 <- t(data.frame(t(sapply(list4,c))))

> corrdf4

activate

bsnl 1.00

call 1.00

pack 1.00

superstar 1.00

please 0.95

tournament 0.85

wimbledon 0.85

enjoy 0.62

experience 0.56

Association of the term "killing" with other terms

> list5<- findAssocs(tdm, "killing", 0.5)

> corrdf5 <- t(data.frame(t(sapply(list5,c))))

> corrdf5

killing

interruptions 0.99

failure 0.98

nzvind 0.98

technical 0.93

cwc 0.83

Association of the term "watch" with other terms

> list6<- findAssocs(tdm, "watch", 0.3)

> corrdf6 <- t(data.frame(t(sapply(list6,c))))

> corrdf6

watch

worldcup 0.48

indvssl 0.40

indvsnz 0.37

**Inference :**

1. Lot of negative reactions of users captured. Users complained about Technical fault, rain

interruptions during CWC and Newzeland vs. India match. Value from corrdf5, corrdf2 and

corrdf1 explain the magnitude of frustration. Undoubtedly Hotstar has been one of the most

popular mediums during this session of sports. People enjoyed a lot. This is explained by corrdf2 and corrdf3.

1. Looks like other programs such as TV serials, movies in Hotstar was not in people's mind or discussion during this time. CWC, Wimbledon dominated during this period.corrdf4 and

corrdf6 quite explain this.

1. BSNL started giving free Hotstar premium subscription with 300GB Superstar pack. We can

find lot of retweet and tweet about this. corrdf3 explains this.

<https://gadgets.ndtv.com/telecom/news/bsnl-superstar-300-broadband-plan-price-rs-749-50-mbps-hotstar-premium-subscription-2057494>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Topic Modeling to identify latent/hidden topics using LDA technique**  > dtm <- as.DocumentTermMatrix(tdm)  > rowTotals <- apply(dtm , 1, sum)  > NullDocs <- dtm[rowTotals==0, ]  > dtm <- dtm[rowTotals> 0, ]  > if (length(NullDocs$dimnames$Docs) > 0) {  + tweets.df <- tweets.df[-as.numeric(NullDocs$dimnames$Docs),]  + }   |  | | --- | | > lda <- LDA(dtm, k = 5) # find 5 topic  > term <- terms(lda, 7) # first 7 terms of every topic  > (term <- apply(term, MARGIN = 2, paste, collapse = ", "))  Topic 1  "please, pack, superstar, activate, bsnl, call, tournament"  Topic 2  "indvnz, india, watching, cwc, match, meanwhile, coz"  Topic 3  "netflix, subscription, today, even, indiavsnewzealand, india, whats"  Topic 4  "watch, streaming, live, worldcup, indvsnz, million, world"  Topic 5  "cwc, technical, failure, interruptions, nzvind, killing, day"  > topics<- topics(lda)  > topics<- data.frame(date=(tweets.df$created), topic = topics) | |  | | |  | | --- | | > qplot (date, ..count.., data=topics, geom ="density", fill= term[topic], position="stack",main="Topic Frequency") | |   F:\Web & Social Media Analytics\GA\Submission plots\Rplot12.png |
| **Inference:**  CWC has been the most discussed topic and lot of negative reaction captured with higher frequencies.  This is something business should think about. How they should act towards customer satisfaction by doing  a root cause analysis of technical glitches, failure and interruptions. Besides this people also enjoyed this  season of sports a lot. They had a great fun. BSNL superstar pack has been also talked about a lot with their  Innovative free hotstar subscription. |
| |  | | --- | |  | |

**Sentiment Analysis:** Understanding emotional valence in tweets using syuzhet

> mysentiment<-get\_nrc\_sentiment((tweets.df$text))

# Get the sentiment score for each emotion

> mysentiment.positive =sum(mysentiment$positive)

> mysentiment.anger =sum(mysentiment$anger)

> mysentiment.anticipation =sum(mysentiment$anticipation)

> mysentiment.disgust =sum(mysentiment$disgust)

> mysentiment.fear =sum(mysentiment$fear)

> mysentiment.joy =sum(mysentiment$joy)

> mysentiment.sadness =sum(mysentiment$sadness)

> mysentiment.surprise =sum(mysentiment$surprise)

> mysentiment.trust =sum(mysentiment$trust)

> mysentiment.negative =sum(mysentiment$negative)

**Presentation of Emotional valance in a bar chart**

# Create the bar chart

> yAxis <- c(mysentiment.positive,

+ mysentiment.anger,

+ mysentiment.anticipation,

+ mysentiment.disgust,

+ mysentiment.fear,

+ mysentiment.joy,

+ mysentiment.sadness,

+ mysentiment.surprise,

+ mysentiment.trust,

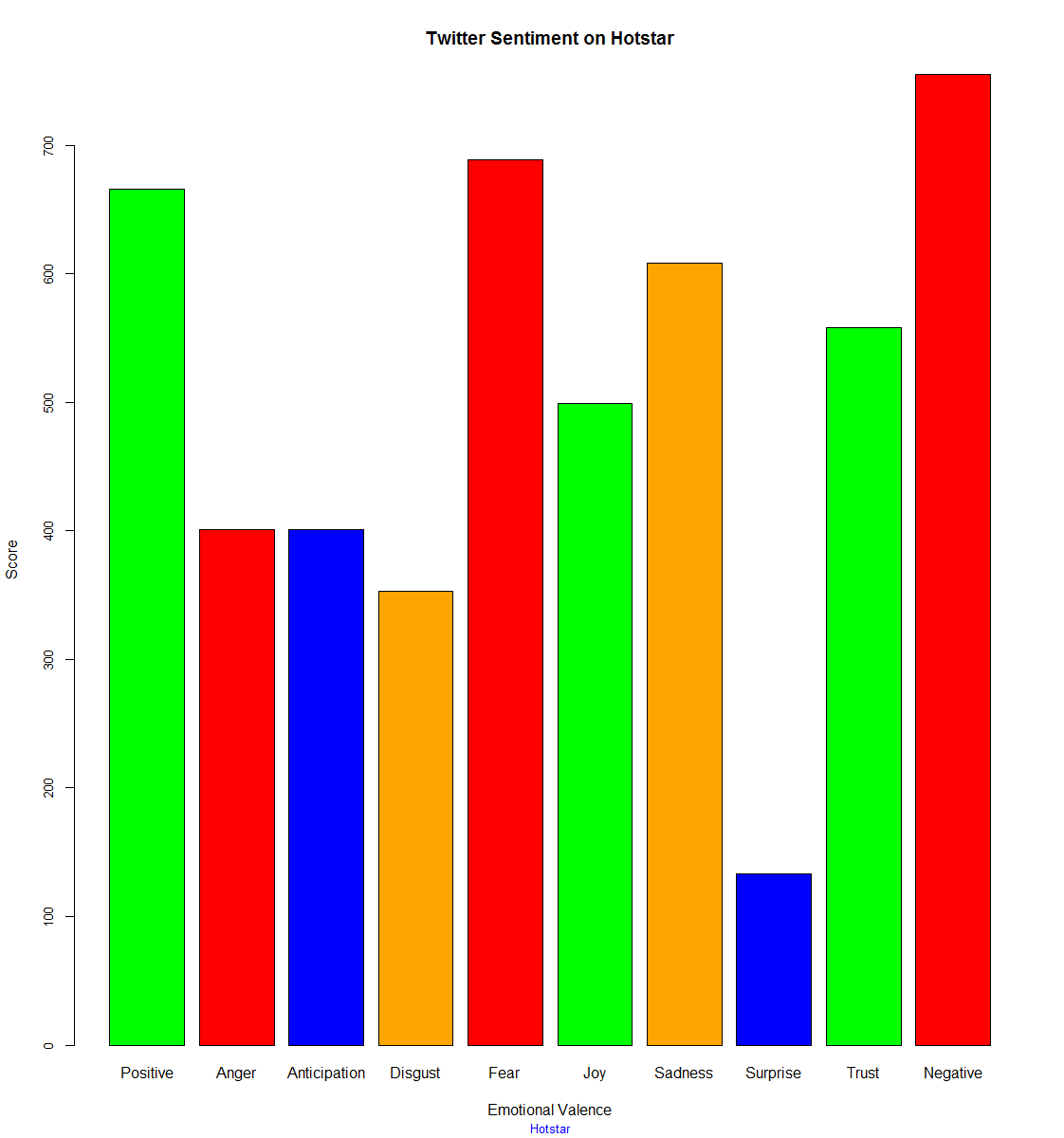
+ mysentiment.negative)

> xAxis <- c("Positive","Anger","Anticipation","Disgust","Fear","Joy","Sadness","Surprise","Trust","Negative")

> colors <- c("green","red","blue","orange","red","green","orange","blue","green","red")

> yRange <- range(0,yAxis)

> barplot(yAxis, names.arg = xAxis, xlab = "Emotional Valence", ylab = "Score", main = "Twitter Sentiment on Hotstar", sub = "Hotstar", col = colors, border = "black", xpd = F, ylim = yRange, axisnames = T, cex.axis = 0.8, cex.sub = 0.8, col.sub = "blue")



**Inference :**

> mysentiment.positive

[1] 666

> mysentiment.anger

[1] 401

> mysentiment.anticipation

[1] 401

> mysentiment.disgust

[1] 353

> mysentiment.fear

[1] 689

> mysentiment.joy

[1] 499

> mysentiment.sadness

[1] 608

> mysentiment.surprise

[1] 133

> mysentiment.trust

[1] 558

> mysentiment.negative

[1] 755

|  |  |
| --- | --- |
| **Sentiment** | **Count** |
| Positive | 666 |
| Anger | 401 |
| Disgust | 353 |
| Fear | 689 |
| Joy | 499 |
| Sadness | 608 |
| Surprise | 133 |
| Trust | 558 |
| Negative | 755 |

* We can see that number of negative Tweets is the highest. This was due to technical glitches HotStar faced due to sudden increase of viewership during CWC semifinal match #IndVsNewZealand.
* Fear, anger, sadness, disgust emotions are also significant in number. This captured

emotions of hotstar viewers during the live broadcast of #IndVsNewZealand match.

* Among the tweets joy, trust and positive emotions are also significant. Although

there were negative reactions of users, but hotstar was able to maintain quality of service.

**Sentiment Analysis : Plot by date - understanding cumulative sentiment score movement**

> mysentimentvalues <- data.frame(get\_sentiment((tweets.df$text)))

> colnames(mysentimentvalues)<-"Popularity"

> mysentimentvalues$Date <- tweets.df$created

> result <- aggregate(Popularity ~ Date, data = mysentimentvalues, sum)

> result

Date Popularity

1 2019-07-05 72.50

2 2019-07-06 111.50

3 2019-07-07 20.20

4 2019-07-08 58.40

**5 2019-07-09 -381.40 << This is the day of IND vs NZ match**

6 2019-07-10 0.55

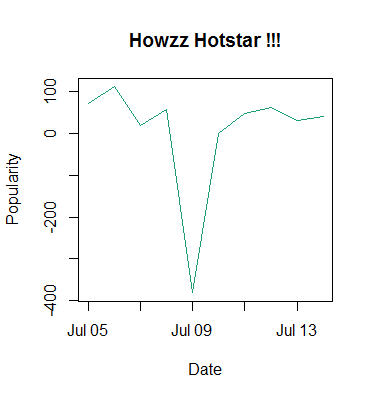
7 2019-07-11 47.35

8 2019-07-12 61.30

9 2019-07-13 31.55

10 2019-07-14 40.10

> plot(result, type = "l",col=brewer.pal(8, "Dark2"), main = "Howzz Hotstar !!!")



The line graph shows Hotstar popularity went down sharply on 9th July 2019, the day of India vs. Newzeland match. We can see that lot of negative tweets posted on that day related to Technical glitches, interruptions etc. Hotstar faced sudden surge of viewership which caused technical issues. On next day Hotstar witnessed highest viewership ever on any live broadcast event – 25.1M. It is recommended to find proper root cause analysis of such technical glitches so that HotStar can handle such sudden surge in viewership on demand anytime.

# Conclusion:

The period for which data has been collected from twitter is one of the most significant periods in the world of sports. Users' expectation was very high, the excitement was in the air. The time was challenging for a medium like hotstar. They had to deliver the best. There were difficulties, there were disappointments but looking at the line graph we can say Hotstar could manage to deliver contents online and people have loved it watching through HotStar medium. Hotstar has now established itself as a serious medium content provider eclipsing other competitors in viewership volume. #CWC19 and #Wimbledon are now synomous with the brand Hotstar in India.

# Appendix:

The R code for the entire analysis :