TWITTER SENTIMENTAL ANALYSIS FOR CRICKET DATASET

Prepared For

## SOCIAL INFORMATION NETWORKS

Submitted To

## SIVA SHANMUGAM G

## Slot

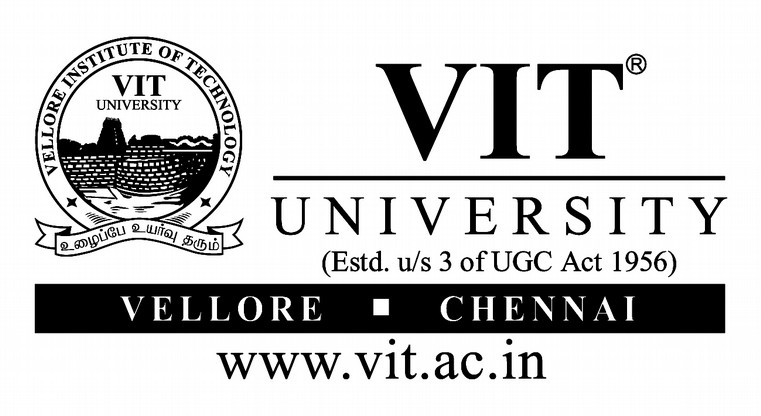
## A2 + TA2

TEAM MEMBERS:-

**16BCE0223 – SR NAVYA SREE**

**15BCE0827-TASNEEM ALLUR FARHANAZ**

**School of Computer Science and Engineering**



**INTRODUCTION:-**

* Numerous outlets available for individuals to express opinions and

emotions...positive, negative, and neutral.

* Need to promote positive news, react to the negative, and move the needle

favorably on neutral news....as near real-time as possible

* Mining high volume, high velocity data for meaningful insights is not

easy!

* Similar challenges exist across all industries/verticals

**ANALYTICS:-**

* What is trending positively/negatively over a period of time and why?
* Who is being talked about, where, and why?
* What college is being talked about?
* What topics are being discussed the most?
* Who is being talked about most positively?
* What are the best sources for positive exposure?
* What is the geographic location of the comments?

**TWITTER SENTIMENT ANALYSIS:-**

Twitter is an online news and social networking service that enables users to send

and read short 140-character messages called "tweets". Registered users can read

and post tweets, but those who are unregistered can only read them.

Hence Twitter is a public platform with a mine of public opinion of people all over

the world and of all age categories.

As of October 2016, Twitter has more than 315 million monthly active users .

Twitter Sentiment Analysis is the process of determining the emotional tone behind a

series of words, used to gain an understanding of the the attitudes, opinions and

emotions expressed within an online mention.

**ABSTRACT:-**

The growth in micro-blogging activity on sites over the last few years has been phenomenal.  Platforms like Twitter offer an easy outlet for people to express their opinions and companies are increasingly getting interested in capturing these insights about customer behaviour and preferences that could help generate more revenues. The staggering amount of data that these sites generate cannot be manually analysed. Enter thus, Sentiment Analysis, the field where we teach machines to understand human sentiment.

In this project, we have proposed some of the Indian players for WorldCup2019 based on the tweets . Tweets are collected and refined and taken for sentiment analysis .The analysis is based upon Naive bayes algorithm. This has been applied on a few Indian cricketers and compared them against each other. Thus we can predict the players according to the positivity of their game in people and analysts

**EXISTING SYSTEM:-**

Sentiment analysis is in demand because of its efficiency. Thousands of text

documents can be processed for sentiment in seconds, compared to the hours it

would take a team of people to manually complete. Because it is so efficient

many businesses are adopting text and sentiment analysis and incorporating it into their processes.

**PROPOSED SYSTEM:-**

On Twitter however, people post 140-character long informal messages called tweets. Analysing sentiment from these tiny pieces of text is challenging due to their unstructured nature- internet slang, abbreviations, non-conventional spelling and grammar, hashtags, urls and emoticons are just some of the complexities that need to be addressed.

Cricket data visualization is the presentation of data in a pictorial or graphical format. It enables decision makers to see analytics presented visually, so they can grasp difficult concepts or identify new patterns. With interactive visualization, you can take the concept a step further by using technology to drill down into charts and

graphs for more detail, interactively changing what data you see and how it’s

processed. Tables, bar plots, timelines, word clouds, histograms and pie charts can be used for visualization. Finally to predict which players are selected to the final world cup team using the collected data.

**APPLICATIONS:-**

The applications of sentiment analysis are broad and powerful. Shifts in sentiment

on social media have been shown to correlate with shifts in the stock market.

For example, the Obama administration used sentiment analysis to gauge public

opinion to policy announcements and campaign messages ahead of 2012 presidential

election.

The ability to quickly understand consumer attitudes and react accordingly is

something that Expedia Canada took advantage of when they noticed that there was

a steady increase in negative feedback to the music used in one of their television

adverts.

**LIMITATIONS:-**

1. The Twitter Search API can get tweets upto a maximum of 7 days old.

2. Not effective in detecting sarcasm.

3. Cannot get 100% efficiency in analysing sentiment of tweets.

4. Can only retrieve a maximum of 1000 tweets per query without

authenticating via OAuth before receiving a 403 error or timeout.

5. Giving a hash tag under the wrong category will still give results: No error

Message

**SYSTEM REQUIREMENTS**

Installation of R

Twitter Authentication to access API

**BRIEF IDEA OF THE PROCESS INVOLVED**

Tweets are imported using R and the data is cleaned by removing emoticons and

URLs. Lexical Analysis as well as Naive Bayes Classifier is used to predict the

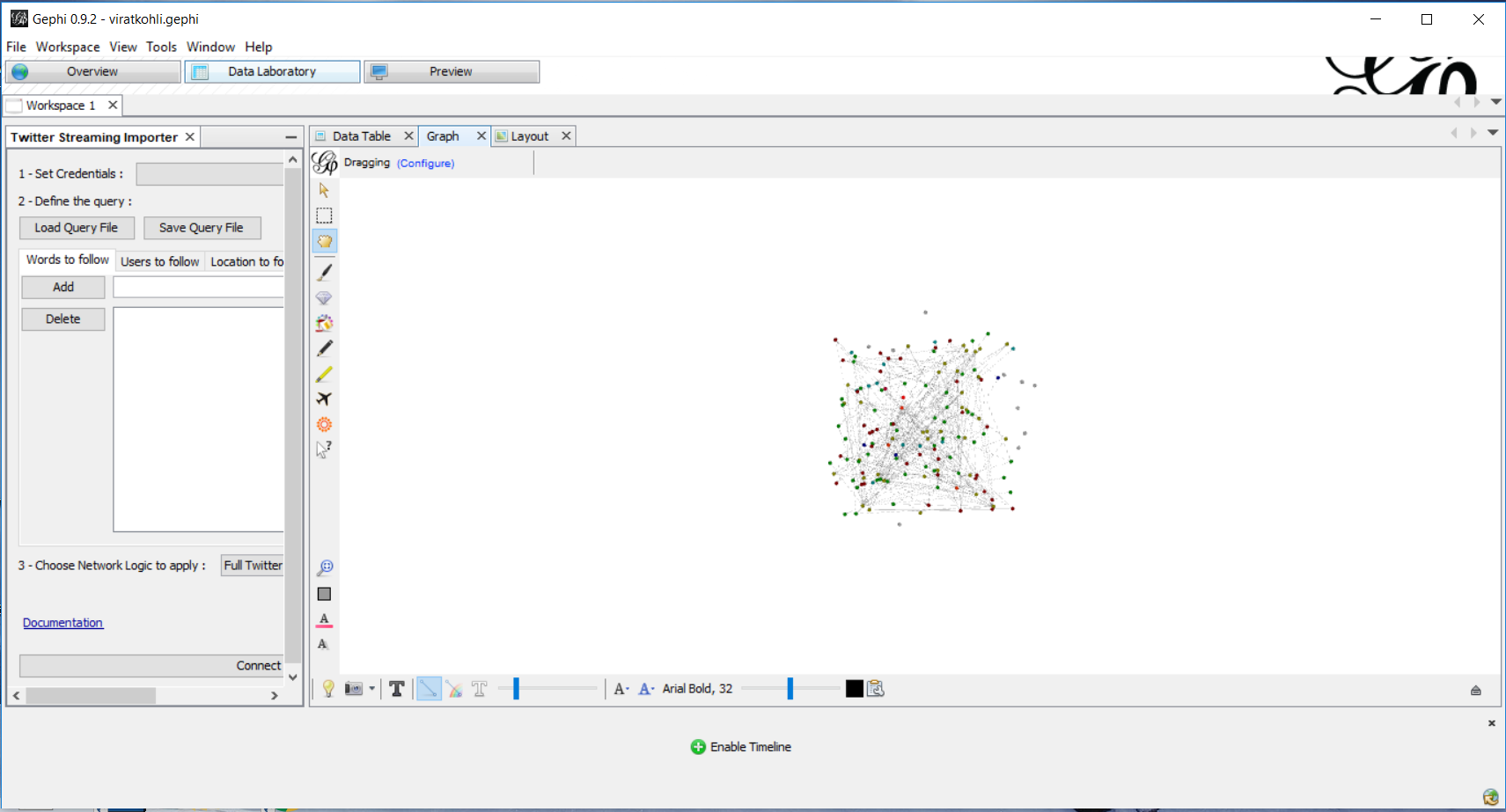
sentiment of tweets and subsequently express the opinion graphically through

histogram, pie chart, wordcloud and tables and finally to predict which players are selected to the final world cup team using the collected data.

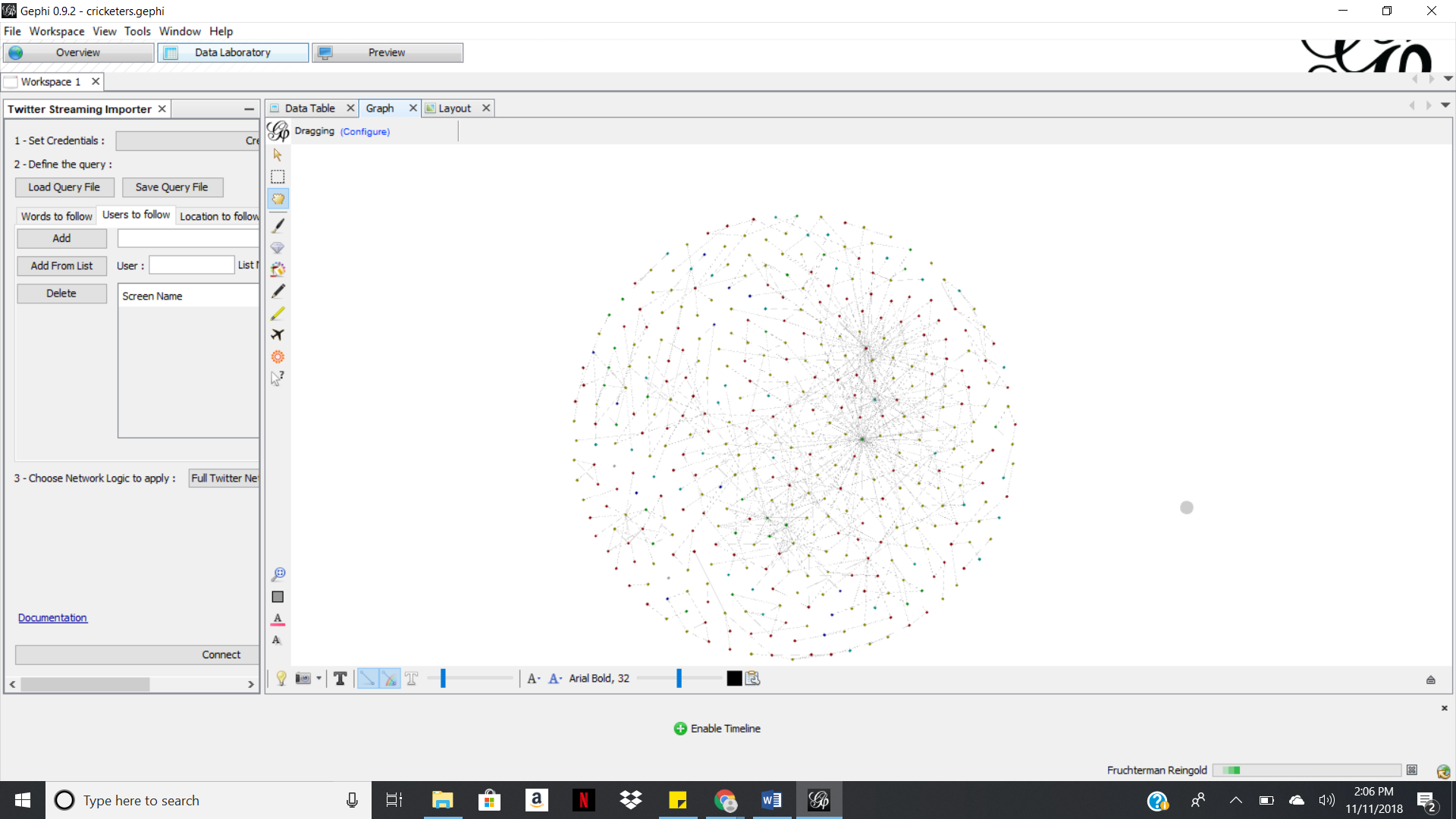
Using gehi :

**We will connect to twitter using credentials**

**#viratkohli**



**#cricketers**



**Steps involved using R**

1. **Extraction of Tweets**

(i) Create twitter application

(ii) twitteR - Provides an interface to the Twitter web API

(iii) ROAuth - R Interface For OAuth

(iv) Create twitter authenticated credential object(using key from step (ii) and

cacert.pem certificate): It is done using consumer key, consumer secret, access

token, access secret.

(v) During authentication, we are redirected to a URL automatically where we

click on Authorize app as shown in the image below and enter the unique

7-digit number to get linked to the account from which feeds are being taken.

1. **Cleaning Tweets**

The tweets are cleaned in R by removing:

* Extra punctuation
* Stop words (Most commonly used words in a language like *the*, *is*, *at*,

*which*, and *on*.)

* Redundant Blank spaces
* Emoticons
* URLS

1. **Algorithms used**

**Lexical Analysis:** By comparing uni-grams to the pre-loaded word

database, the tweet is assigned sentiment score - positive, negative or

neutral and overall score is calculated.

**Naive Bayes Machine Learning Algorithm:** Training data sets are

used to teach the machine what kind of sentences are categorized as

positive and what kind are categorized as negative. On arrival of a new

tweet or sentence, the machine uses this algorithm to give the correct

category to the new data and adds level to the emotion.

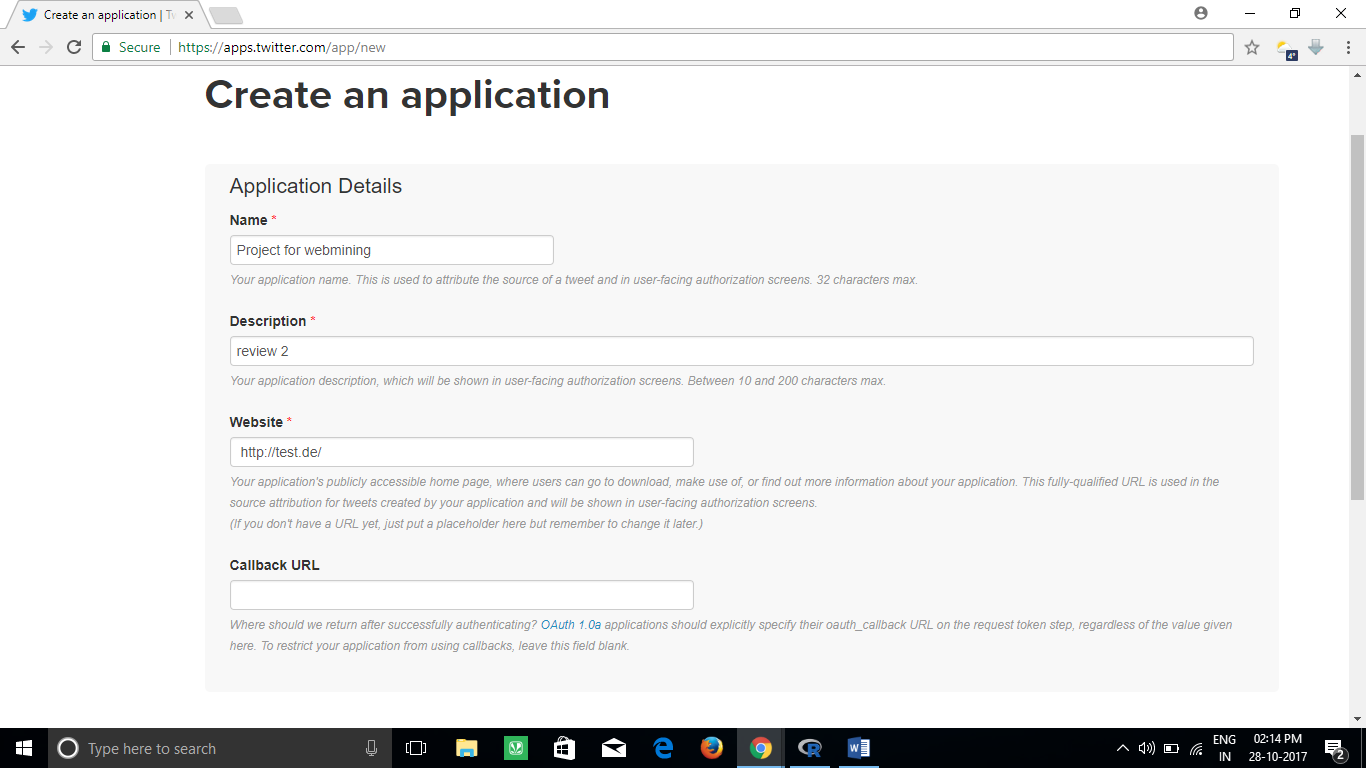
1. **Histogram tab : histogram plot**

Histograms of positive, negative and overall score are found under the Histogram tab

for graphically analyzing the intensity of emotion in the tweeters

STEPS INVOLVED :-

Step 1:-Creating a twitter application



Step 2:- code for authorizing the twitter app with the R software to collect the tweets

# authorisation

if (!require('pacman')) install.packages('pacman')

pacman::p\_load(twitteR, ROAuth, RCurl)

api\_key = 'JlHHunP3gprl9HjWLL9HFS2kh'

api\_secret = '0k0z8fMvIVfdl4xBfCEuk5KQZxxP49OWBTyuTojSAkNxP3ksV1 '

access\_token = '573546348-HAMbhb0MPRr7gMWRu7J0z1V8EW5KI0QxlAM5sQ1S '

access\_token\_secret = ' ghhuhf57UmEgsrYTVKyqsyTvyMTaVlx1CmYLcWiI13HIZ'

# Set SSL certs globally

options(RCurlOptions = list(cainfo = system.file('CurlSSL', 'cacert.pem', package = 'RCurl')))

# set up the URLs

reqURL = 'https://api.twitter.com/oauth/request\_token'

accessURL = 'https://api.twitter.com/oauth/access\_token'

authURL = 'https://api.twitter.com/oauth/authorize'

twitCred = OAuthFactory$new(consumerKey = api\_key, consumerSecret = api\_secret, requestURL = reqURL, accessURL = accessURL, authURL = authURL)

twitCred$handshake(cainfo = system.file('CurlSSL', 'cacert.pem', package = 'RCurl'))

Step 3:- code for installing the sentiment package in the R software:-

if (!require('pacman')) install.packages('pacman&')

pacman::p\_load(devtools, installr)

install.Rtools()

install\_url('<http://cran.r-project.org/src/contrib/Archive/Rstem/Rstem_0.4-1.tar.gz>')

install\_url('<http://cran.r-project.org/src/contrib/Archive/sentiment/sentiment_0.2.tar.gz>')

Step 4:- Creating a script to search twitter:-

if (!require('pacman')) install.packages('pacman')

pacman::p\_load(twitteR, sentiment, plyr, ggplot2, wordcloud, RColorBrewer, httpuv, RCurl, base64enc)

options(RCurlOptions = list(cainfo = system.file('CurlSSL', 'cacert.pem', package = 'RCurl')))

api\_key = 'JlHHunP3gprl9HjWLL9HFS2kh'

api\_secret = '0k0z8fMvIVfdl4xBfCEuk5KQZxxP49OWBTyuTojSAkNxP3ksV1 '

access\_token = '573546348-HAMbhb0MPRr7gMWRu7J0z1V8EW5KI0QxlAM5sQ1S '

access\_token\_secret = ' ghhuhf57UmEgsrYTVKyqsyTvyMTaVlx1CmYLcWiI13HIZ'

setup\_twitter\_oauth(api\_key,api\_secret)

Step 5:- Code for collecting the tweets:-

# harvest some tweets

some\_tweets = searchTwitter('cricket', n=10000, lang='en')

# get the text

some\_txt = sapply(some\_tweets, function(x) x$getText())

Step 6:- Code for Cleaning the tweets for sentimental analysis:-

# remove retweet entities

some\_txt = gsub('(RT|via)((?:\\b\\W\*@\\w+)+)', '', some\_txt)

# remove at people

some\_txt = gsub('@\\w+', '', some\_txt)

# remove punctuation

some\_txt = gsub('[[:punct:]]', '', some\_txt)

# remove numbers

some\_txt = gsub('[[:digit:]]', '', some\_txt)

# remove html links

some\_txt = gsub('http\\w+', '', some\_txt)

# remove unnecessary spaces

some\_txt = gsub('[ \t]{2,}', '', some\_txt)

some\_txt = gsub('^\\s+|\\s+$', '', some\_txt)

# define 'tolower error handling' function

try.error = function(x)

{

# create missing value

y = NA

# tryCatch error

try\_error = tryCatch(tolower(x), error=function(e) e)

# if not an error

if (!inherits(try\_error, 'error'))

y = tolower(x)

# result

return(y)

}

# lower case using try.error with sapply

some\_txt = sapply(some\_txt, try.error)

# remove NAs in some\_txt

some\_txt = some\_txt[!is.na(some\_txt)]

names(some\_txt) = NULL

Step 7:- Code for performing the sentiment analysis:-

# Perform Sentiment Analysis

# classify emotion

class\_emo = classify\_emotion(some\_txt, algorithm='bayes', prior=1.0)

# get emotion best fit

emotion = class\_emo[,7]

# substitute NA's by 'unknown'

emotion[is.na(emotion)] = 'unknown'

# classify polarity

class\_pol = classify\_polarity(some\_txt, algorithm='bayes')

# get polarity best fit

polarity = class\_pol[,4]

# Create data frame with the results and obtain some general statistics

# data frame with results

sent\_df = data.frame(text=some\_txt, emotion=emotion,

polarity=polarity, stringsAsFactors=FALSE)

# sort data frame

sent\_df = within(sent\_df,

emotion <- factor(emotion, levels=names(sort(table(emotion), decreasing=TRUE))))

Step 8:- Code for determining whether the tweet is positive or negative:-

# plot distribution of polarity

ggplot(sent\_df, aes(x=polarity)) +

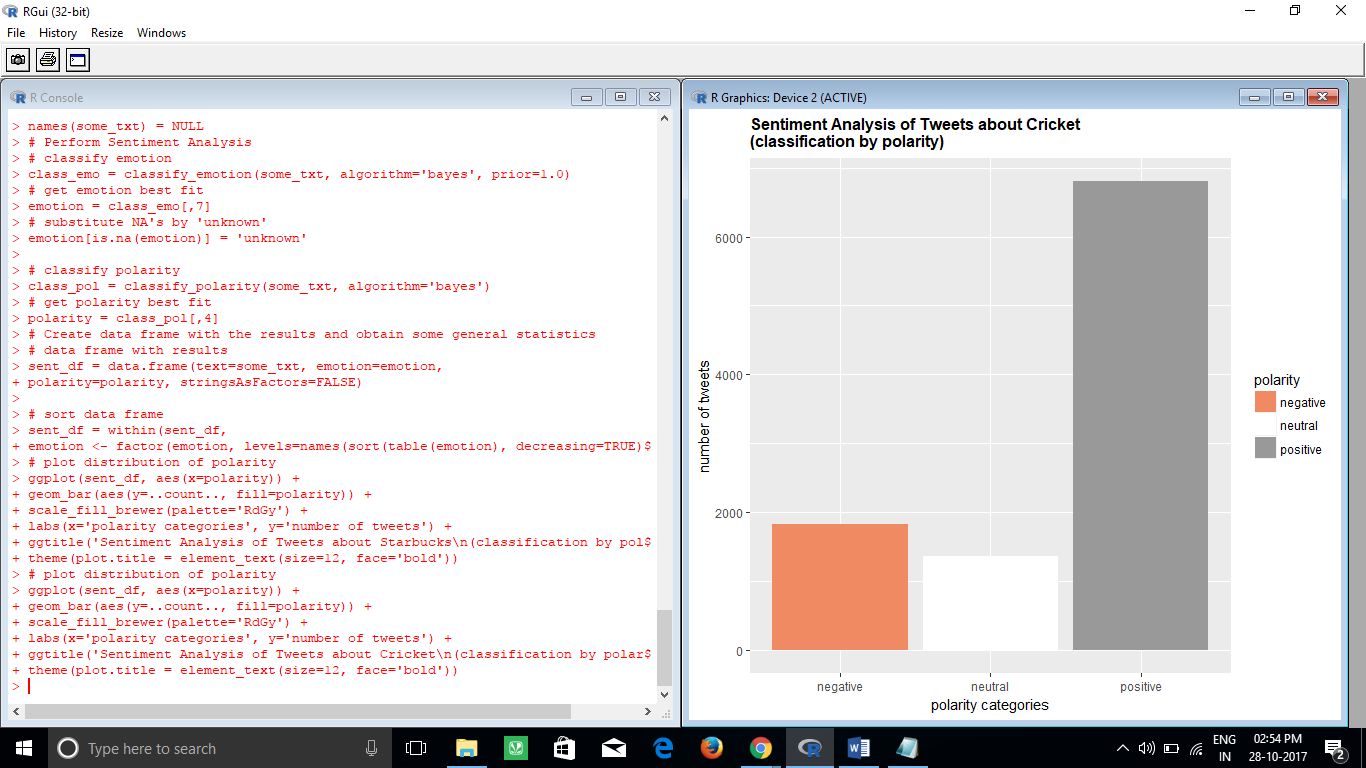
geom\_bar(aes(y=..count.., fill=polarity)) +

scale\_fill\_brewer(palette='RdGy') +

labs(x='polarity categories', y='number of tweets') +

ggtitle('Sentiment Analysis of Tweets about Cricket\n(classification by polarity)') +

theme(plot.title = element\_text(size=12, face='bold'))



Step 9:- Code for producing the histogram:-

# Let’s do some plots of the obtained results

# plot distribution of emotions

ggplot(sent\_df, aes(x=emotion)) +

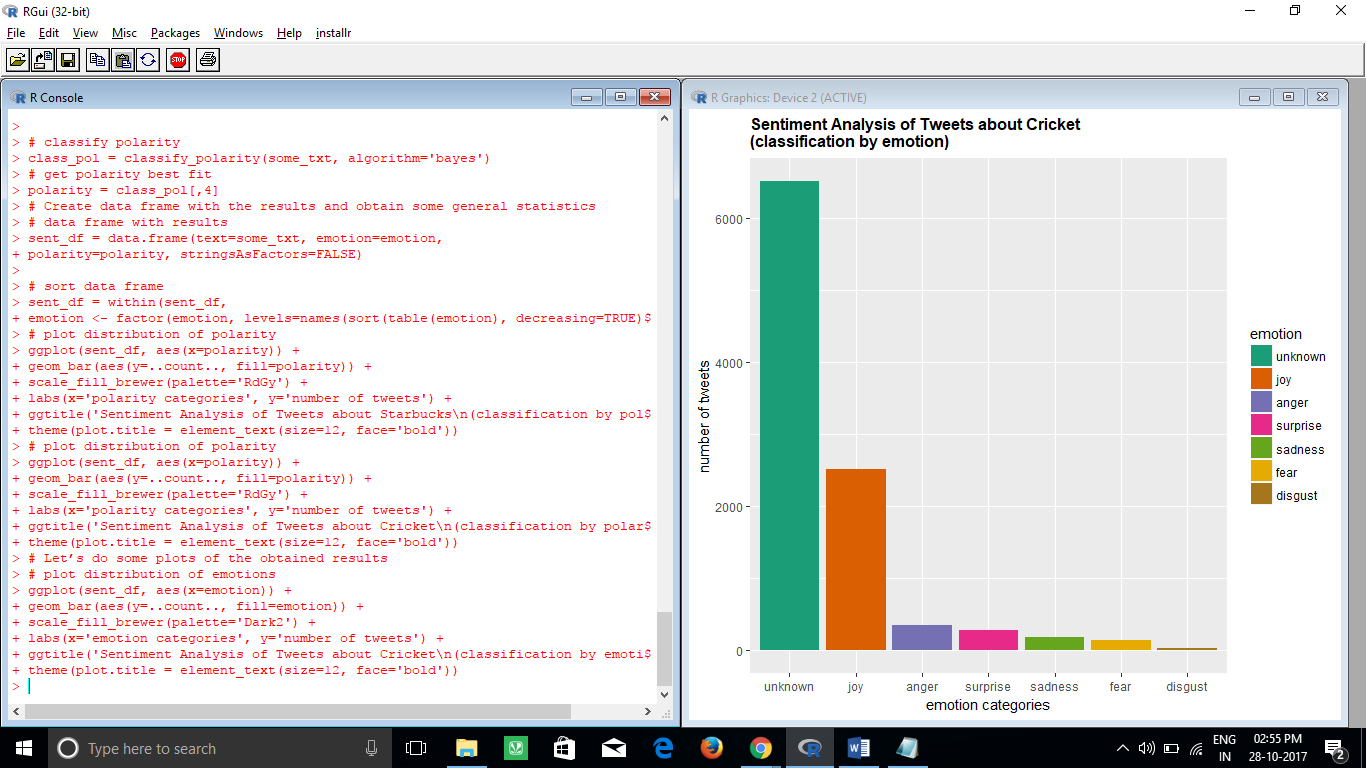
geom\_bar(aes(y=..count.., fill=emotion)) +

scale\_fill\_brewer(palette='Dark2') +

labs(x='emotion categories', y='number of tweets') +

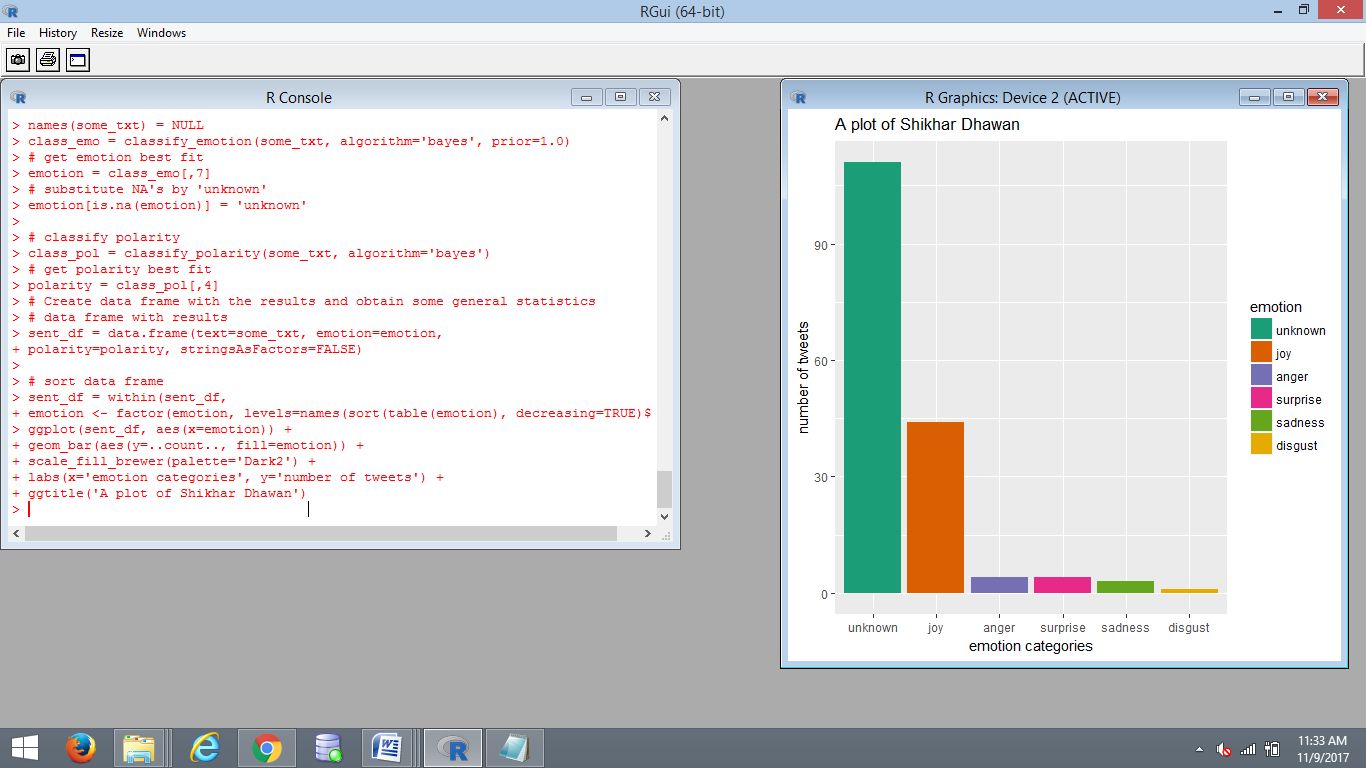
ggtitle('Sentiment Analysis of Tweets about Cricket\n(classification by emotion)') +

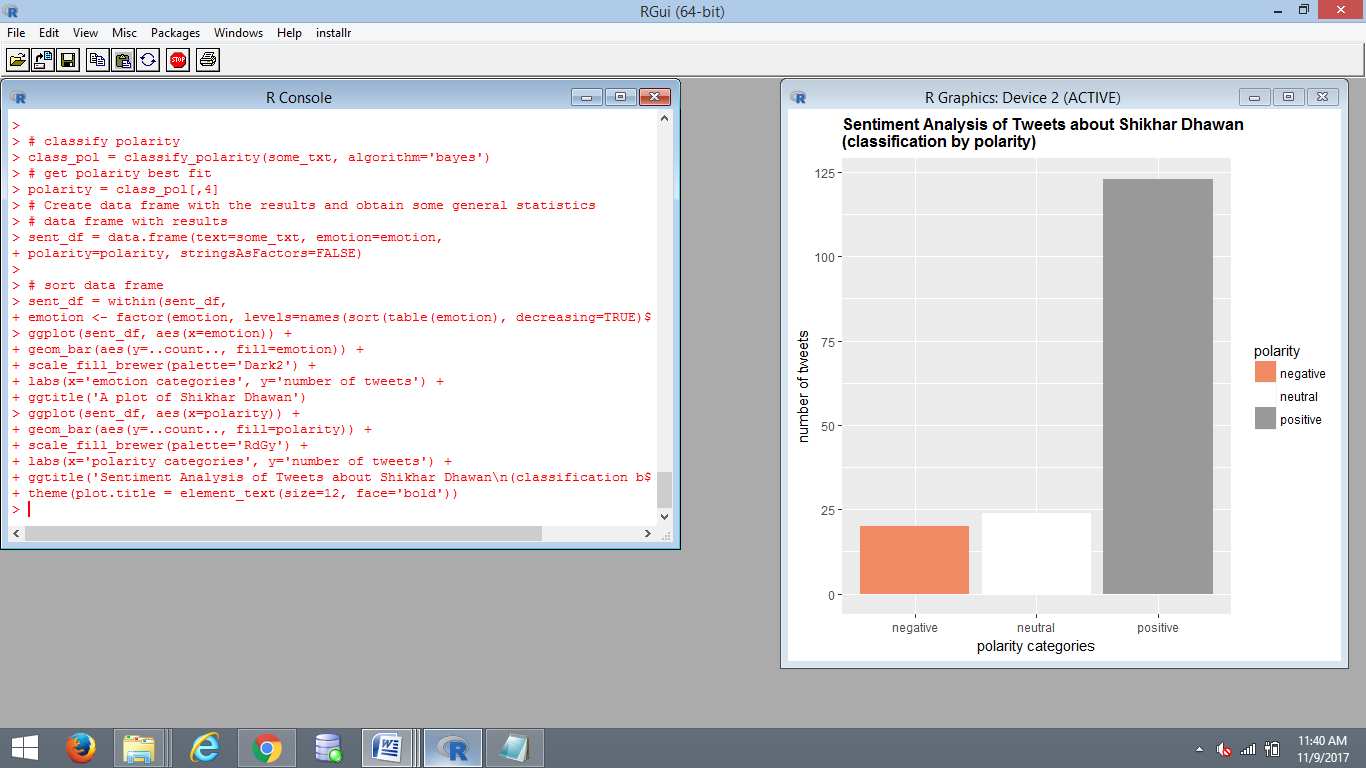
theme(plot.title = element\_text(size=12, face='bold'))



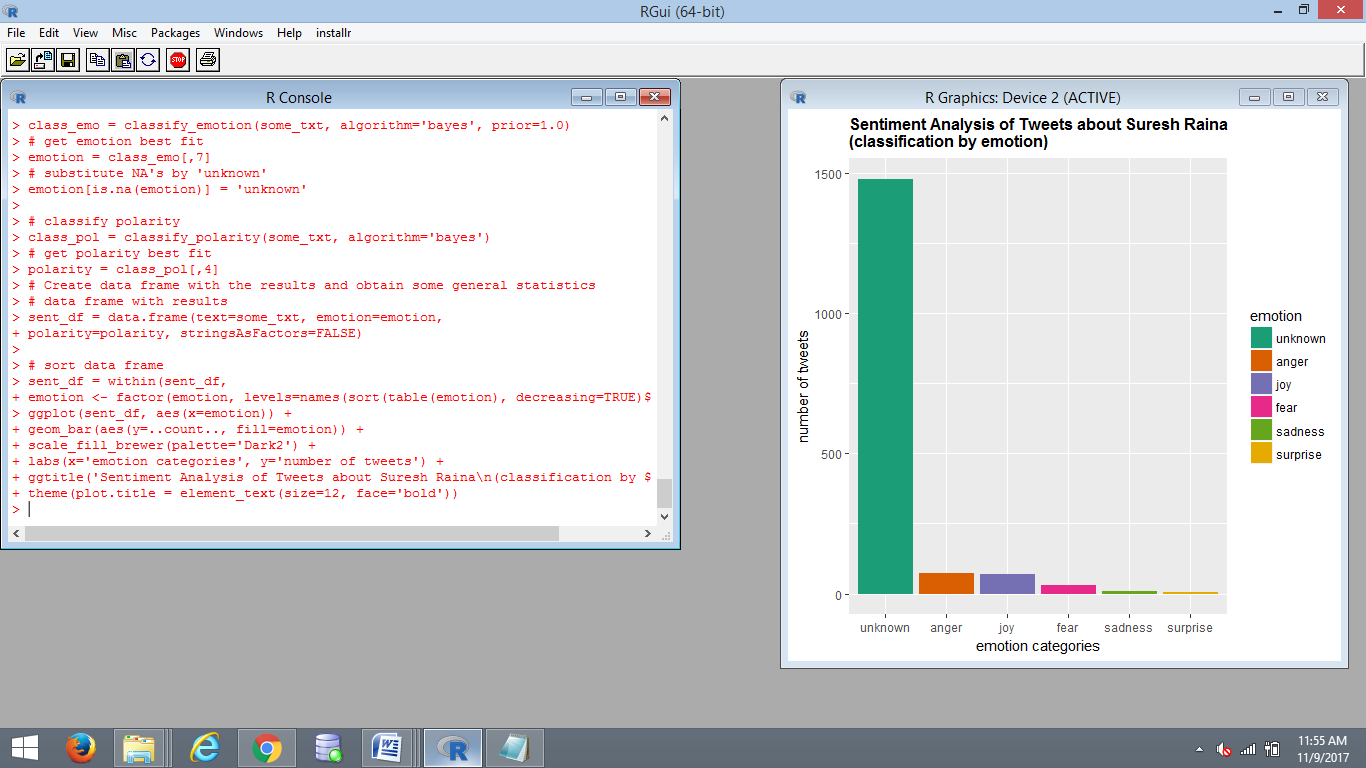
**STEPS PERFORMED ON DIFFERENT CRICKET PLAYERS:-**

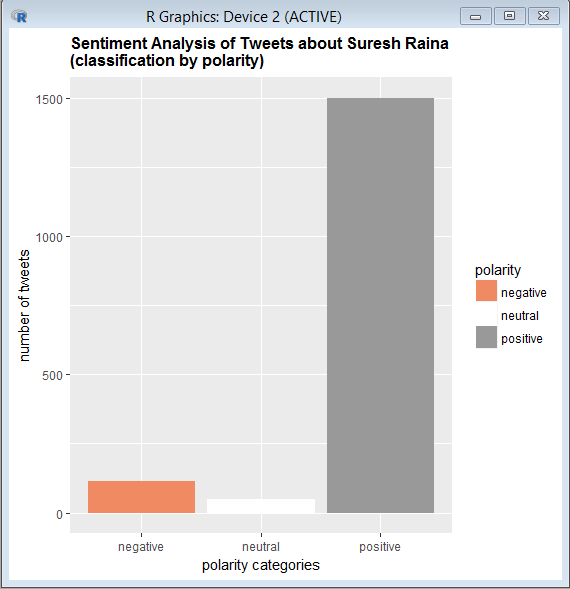
1. **SHIKAR DHAWAN:-**



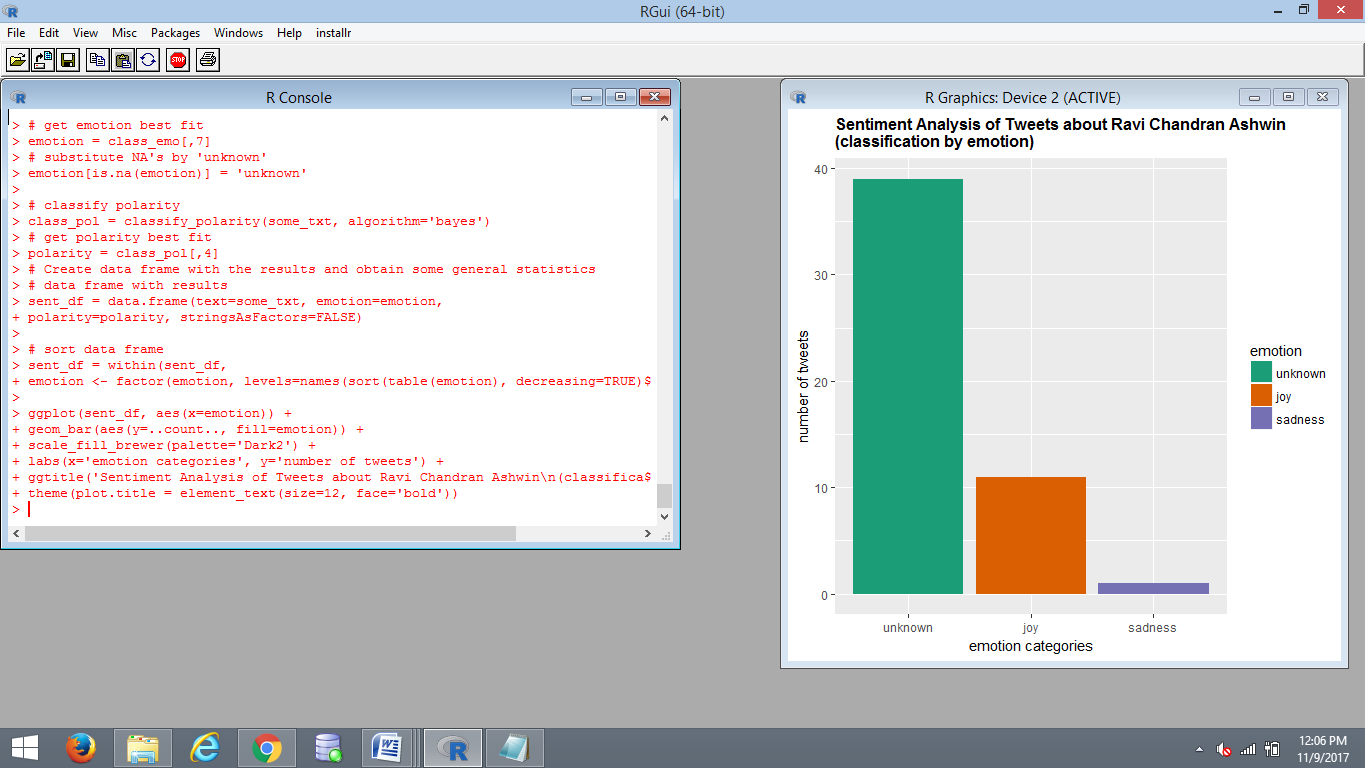


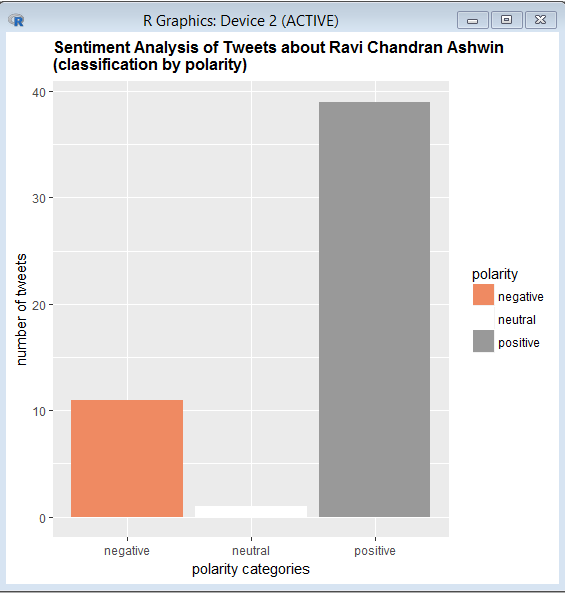
1. **SURESH RAINA:-**



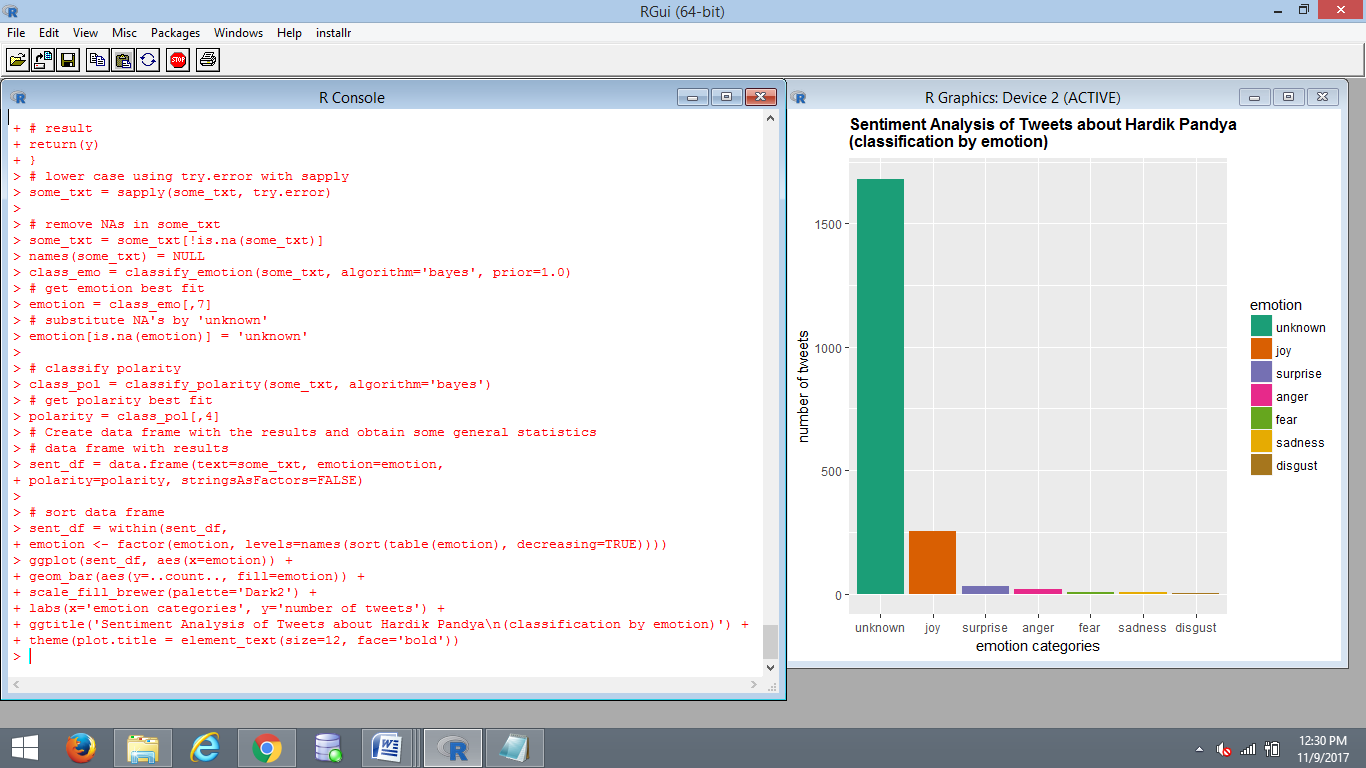


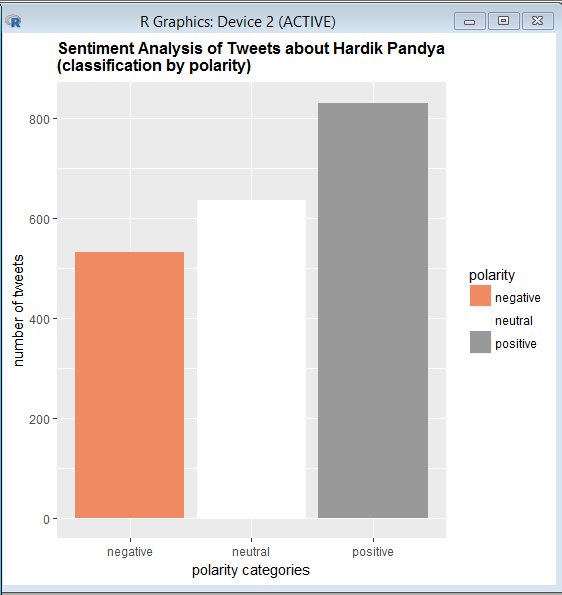
1. **ASHWIN:-**



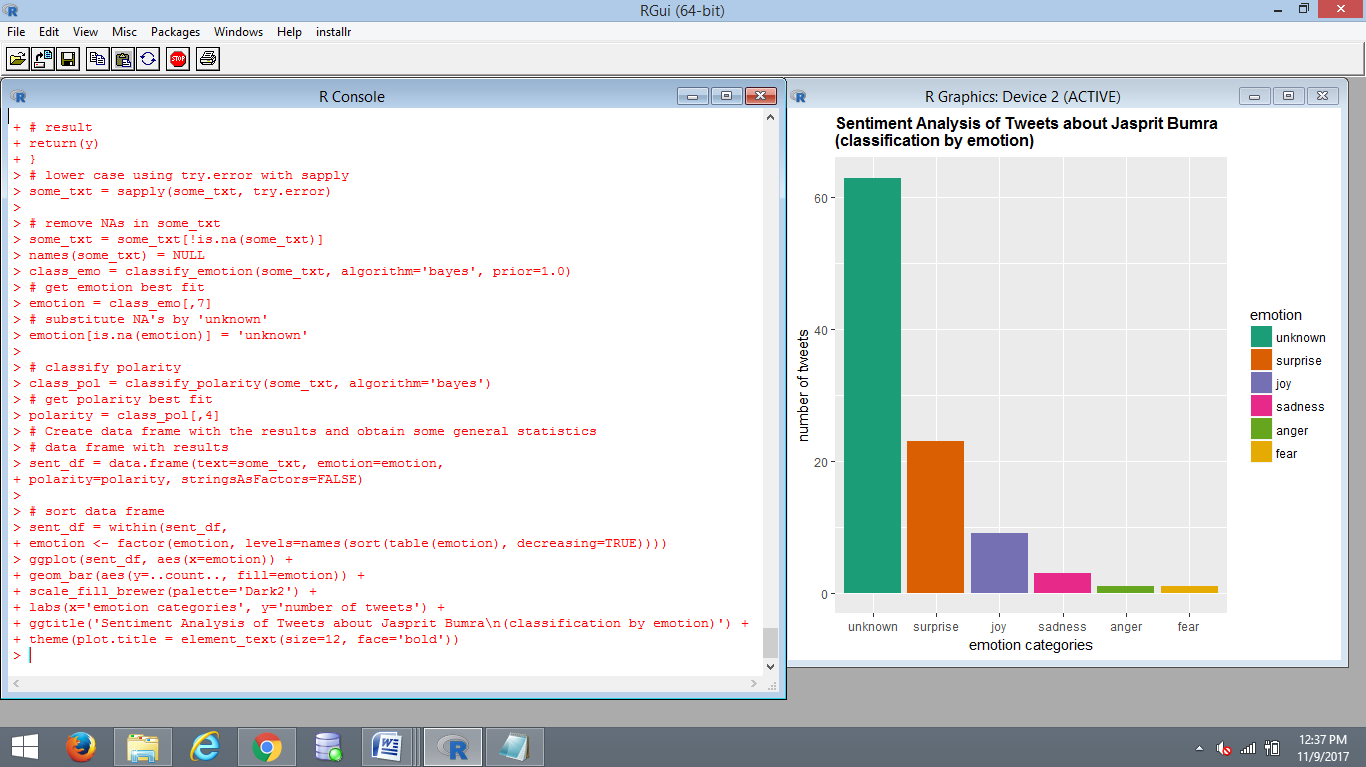


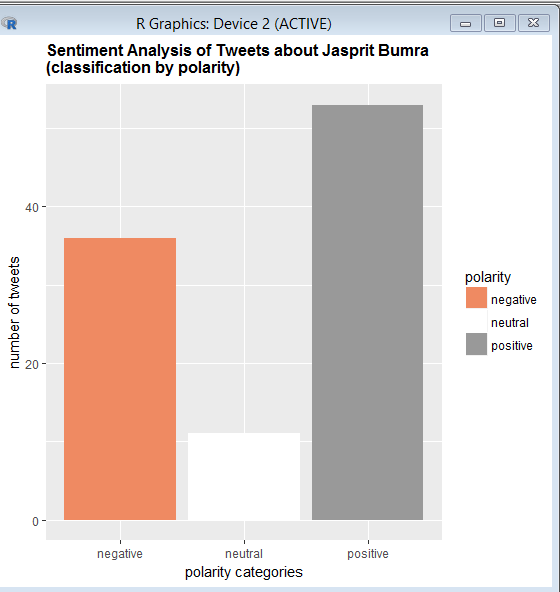
1. **HARDIK PANDYA:-**



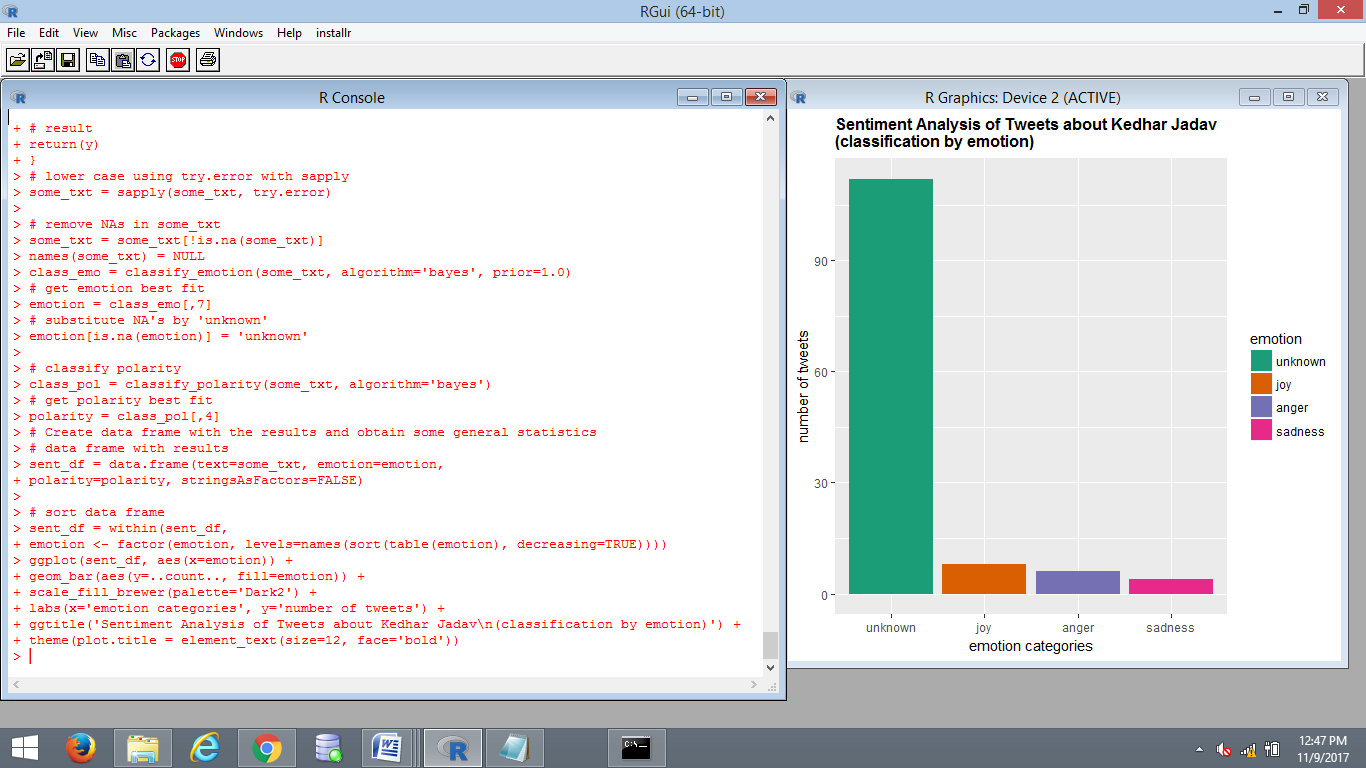


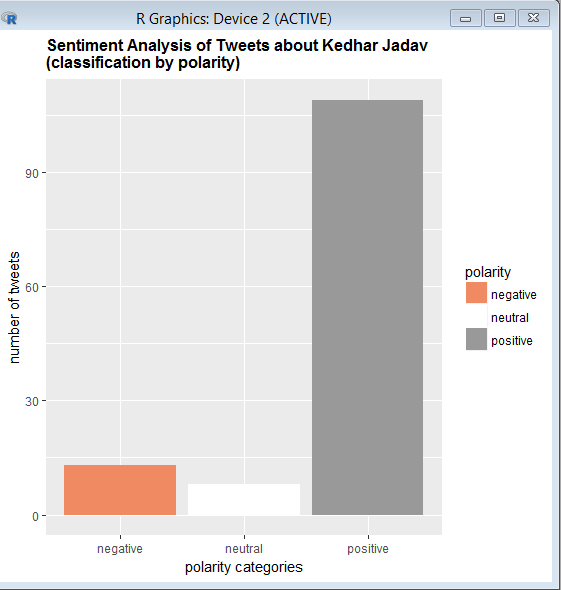
1. **JASPRIT BUMRAH:-**



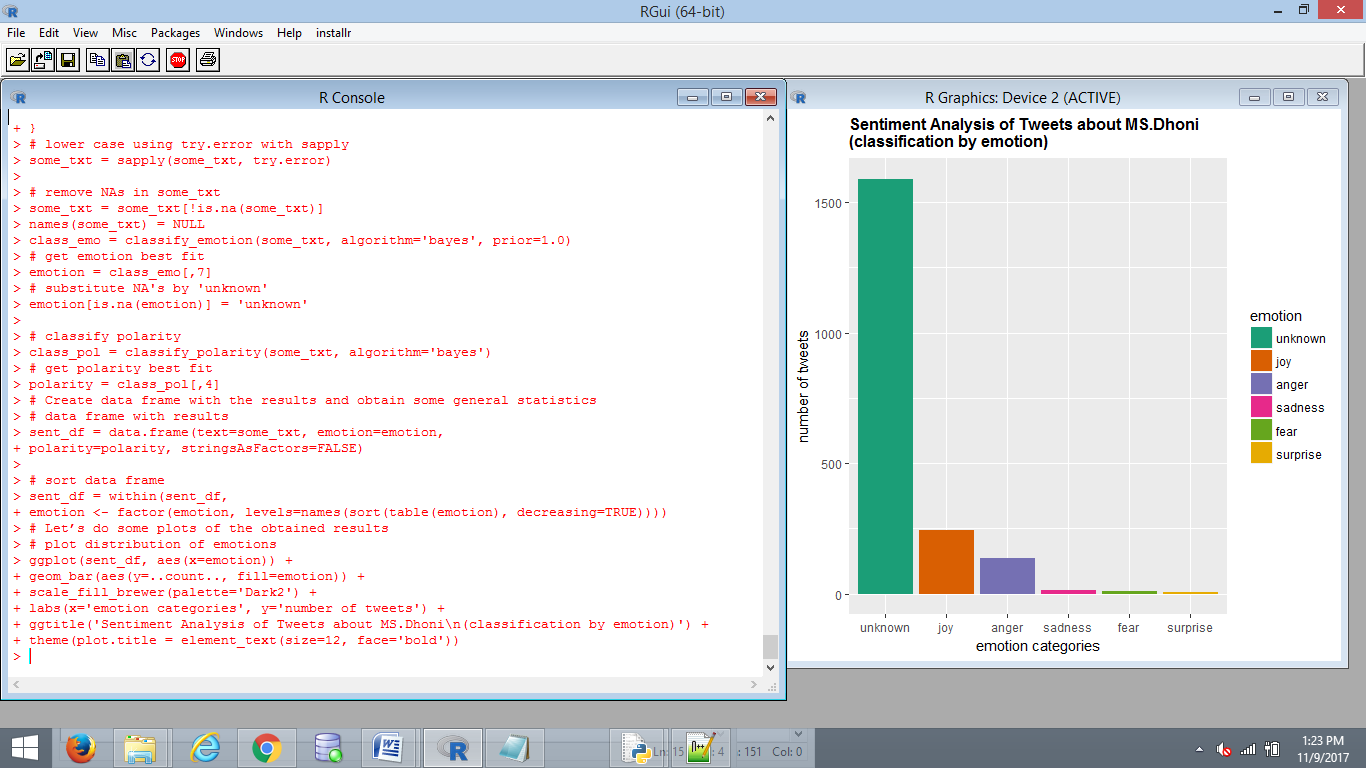


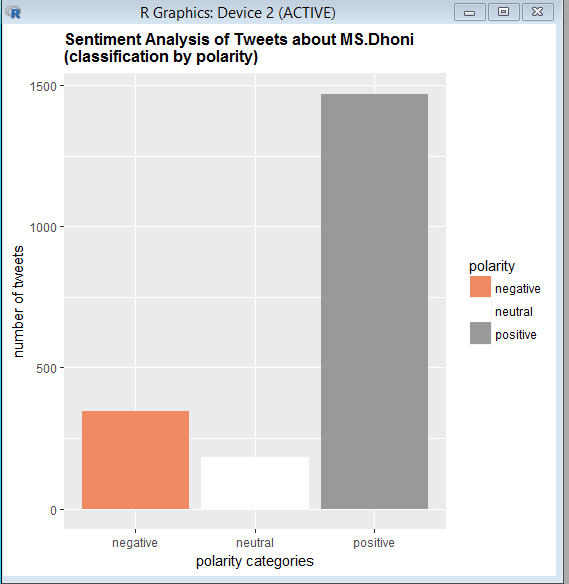
1. **KEDAR JADHAV:-**



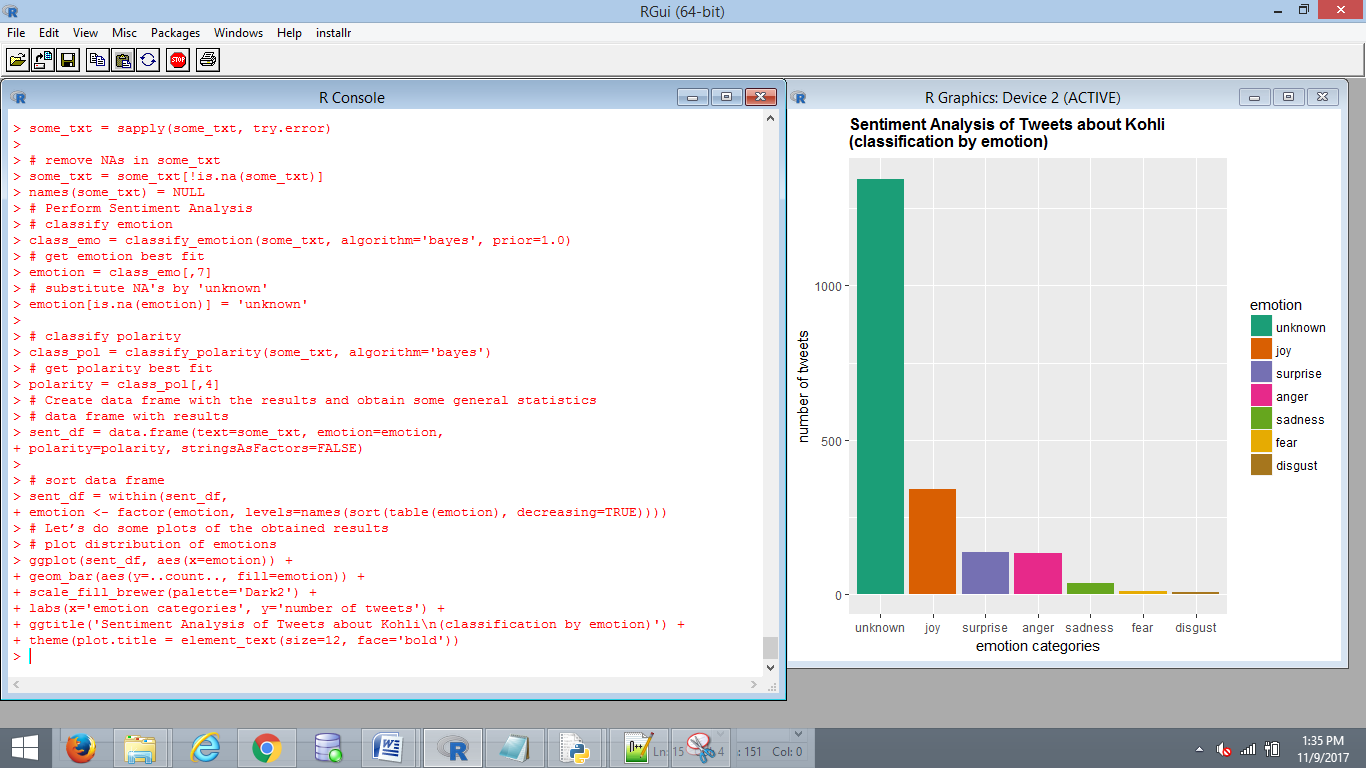


1. **MS DHONI:-**



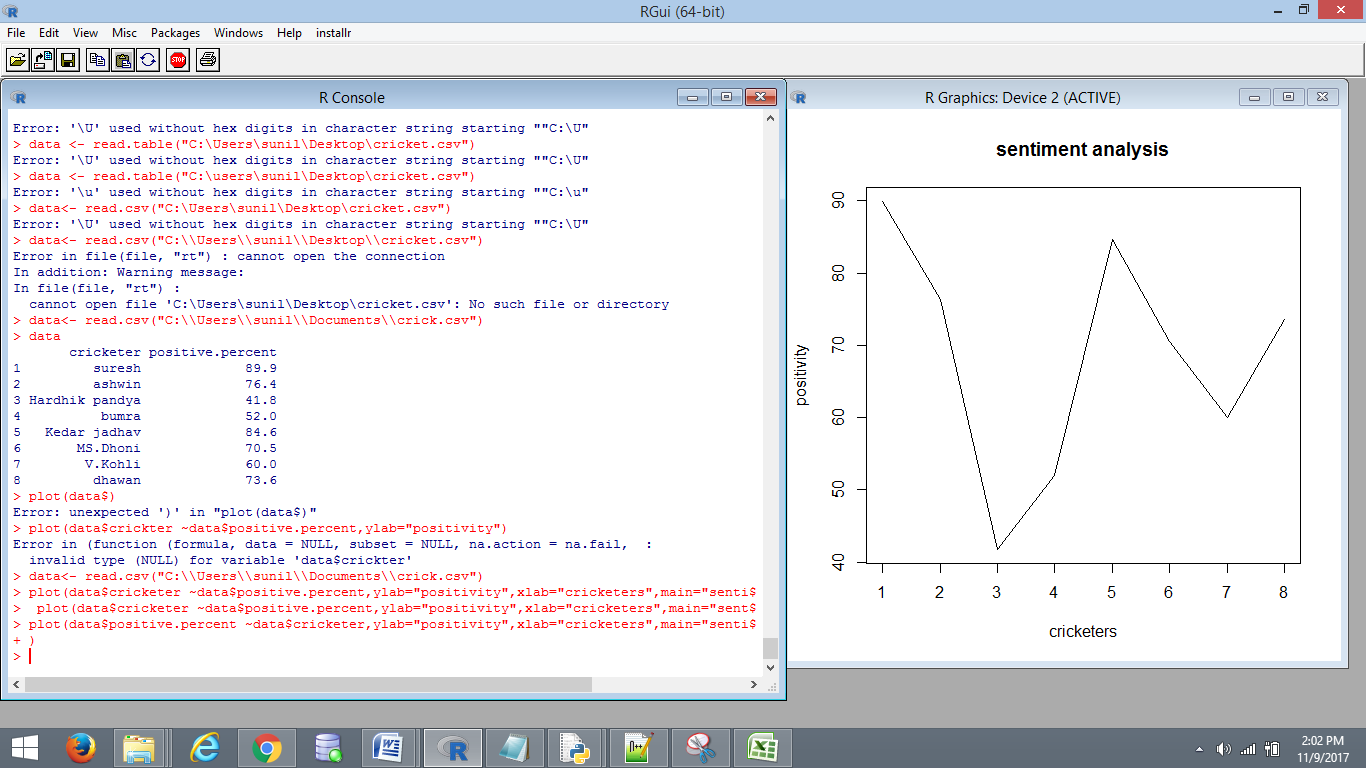


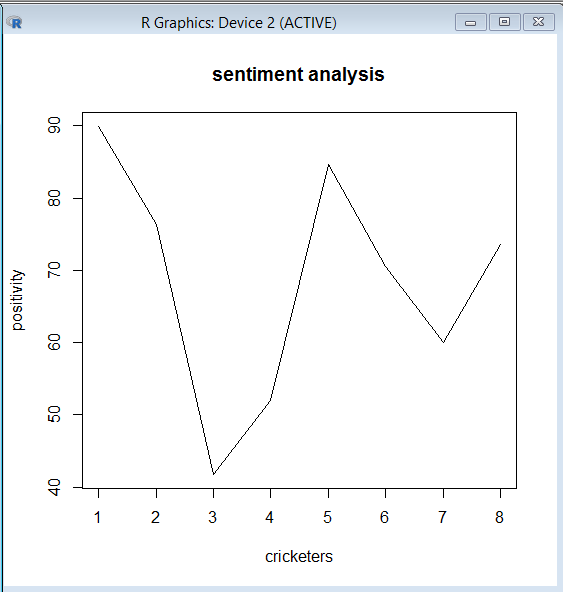
1. **KOHLI:-**



FINAL OUTPUT:-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| cricketers | positive | negative | total | Positivity percent |
| dhawan | 123 | 20 | 167 | 73.6 |
| suresh | 1500 | 110 | 1667 | 89.9 |
| ashwin | 39 | 11 | 51 | 76.4 |
| Hardhik pandya | 820 | 530 | 1960 | 41.8 |
| bumra | 52 | 35 | 100 | 52 |
| Kedar jadhav | 110 | 13 | 130 | 84.6 |
| MS.Dhoni | 1410 | 280 | 2000 | 70.5 |
| V.Kohli | 1200 | 420 | 2000 | 60 |





PLOT INFORMATION:-

1.suresh

2.ashwin

3.hardik

4.bumrah

5.jadhav

6.dhoni

7.kohli

8.dhawan