TWEET BASED INTELLIGENT MOVIE MARKETING

From:

Shreya Jain

Shruti Mandke

Mushtaq Rizvi

MOTIVE

To create an end to end user application for Movie Producers using movie's trailer tweets to analyze the feedback of a trailer (either positive or negative) and to predict the number of screens that can be screened in a particular location.

TWEETS from Different Users

A girl with a camera
@ludoamina

Follow

Skyfall was fantastic. Spoke a lot in the movie though, sorry @aseeqah . It was a stunning movie. @Jaguar was even in it. Love that car.

Reply Ret

1 RETWEET | 1 FAVOR

10:48 AM - 4 Dec 12

Edison Abel
@EdisonAbel

Follow

The Twilight Saga: Breaking Dawn Part 2 th worst movie I ever watched please god save us!!!! No one Dies or any blood anything...

Reply Retweet Favorite

7:36 AM - 3 Dec 12 · Embed this Tweet

Shar
@SharothRocks

Red dawn was awesome! Go wolverines!! @jhutch1992

Reply Retweet Favorite

11:02 AM - 4 Dec

(^_^)
@KatOnYaTongue_

@itsagentP

I wanna watch Rise Of The Guardians... Too bad it's exam week :(

Reply Retweet Favorite

1 RETWEET

Follow

Dons ★
@dhaneeyaa

Follow

Hehe movie marathon with the girls today. Wreck-it ralph and breaking dawn and both were awesome! Great day ♥

Reply Retweet Favorite

6:05 AM - 4 Dec 12 · Embed this Tweet

berly Chia
@kimberly_chia

Follow

Just caught Breaking Dawn Part 2!!! Best out of all the Twilight Saga movies! Got goosebumps watching the epic war scene!

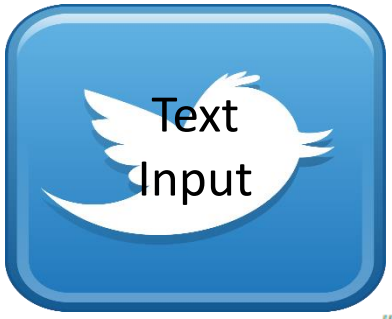
Reply Retweet Favorite

212 RETWEETS | 155 FAVORITES

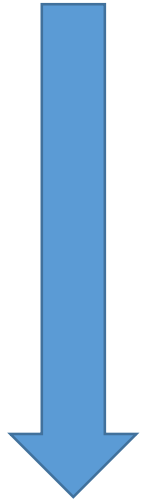
Why Twitter for Sentiment Analysis?

- Popular microblogging site
- Short Text Messages of 140 characters
- 240+ million active users
- 500 million tweets are generated everyday
- Twitter audience varies from common man to celebrities
- Users often discuss current affairs and share personal views on various subjects
- Tweets are small in length and hence unambiguous

DATA EXTRACTION



```
#connection between R Studio and Twitter
key <- "brFgnrk6dFewBOWBiD2m0tANA"
secret <- "kvByOQydL4AIFnMnmAVVYMF4klgSNJwtkNbA5qmTCAisqv6QAT"
secrettk <- "7XfA0v9j0utKeUuf44n2YEB3AtzqVlMM0ue4IrJC0v2cK"
mytoken <- "708481334482698240-QTn0EaokD6IVWFH0ZUhzlW48rdl42Qt"

setup_twitter_oauth(key,secret,mytoken,secrettk)
```



Extraction of
Data from Tweets

```
#using function getText to extract text part of tweets
text <- sapply(captainamericatweets,function(x) x$getText())

#converting latin1 characters to ASCII.
text <- sapply(text,function(row) iconv(row, "latin1", "ASCII", sub = ""))
# remove retweet entities
text = gsub("(RT|via)((?:\\b\\W*@\\w+)+)", "", text)
head(text)
# remove at people
text = gsub("@\\w+", "", text)
# remove punctuation
text = gsub("[[:punct:]]", "", text)
# remove numbers
text = gsub("[[:digit:]]", "", text)
# remove html links
text = gsub("http\\w+", "", text)

#remove captain america civil war from texts
text= gsub("Captain America","",text)
text = gsub("Civil War","",text)
text = gsub("war","",text)
# remove unnecessary spaces
text = gsub("[ \t]{2,}", "", text)
text = gsub("^\\s+|\\s+$", "", text)

# 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
text = sapply(text, try.error)
nrow(text)
# remove NAs in some_txt
#text = text[!is.na(text)]
#names(text) = NULL
```

DATA CLEANING

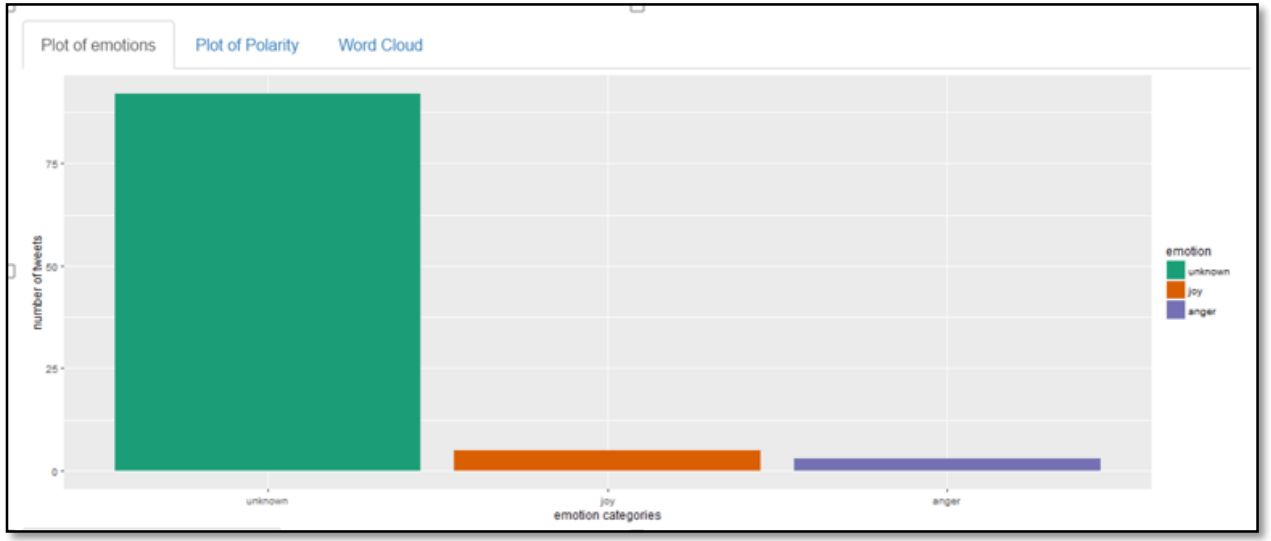
SENTIMENT ANALYSIS

Sentiment analysis is used to see if a text is neutral, positive or negative.

```
> example = c("it is bad", "It is good", "I am batman", "It's expensive and useless")
> examplesentiment = score.sentiment(example,pos,neg)
> examplesentiment
                       text score
1                  it is bad    -1
2                 It is good     1
3                I am batman     0
4 It's expensive and useless    -2
```

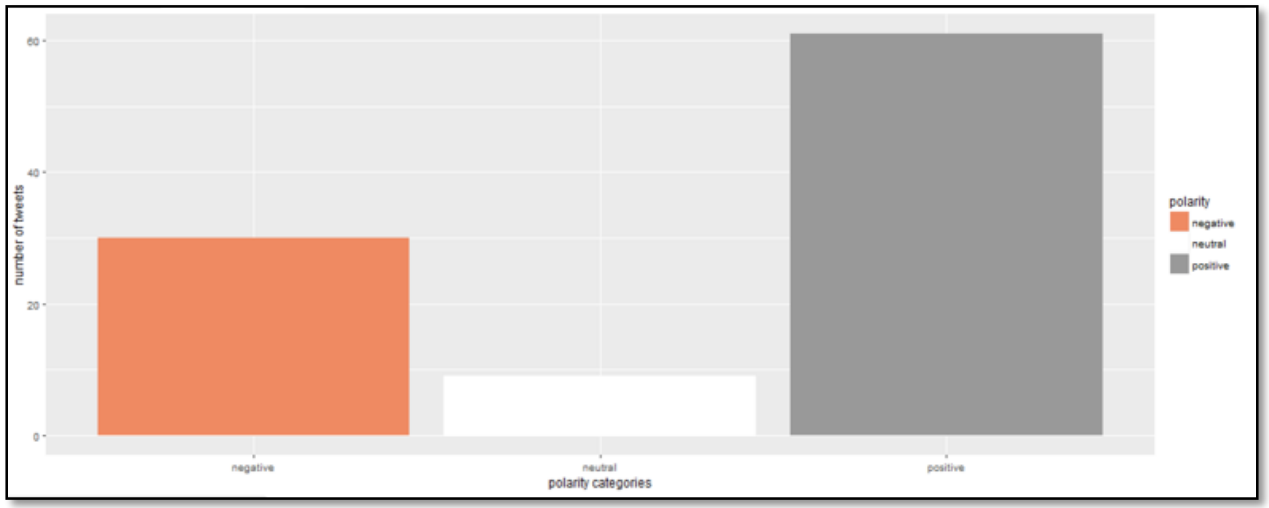
VISUALIZATION – Emotion Plot

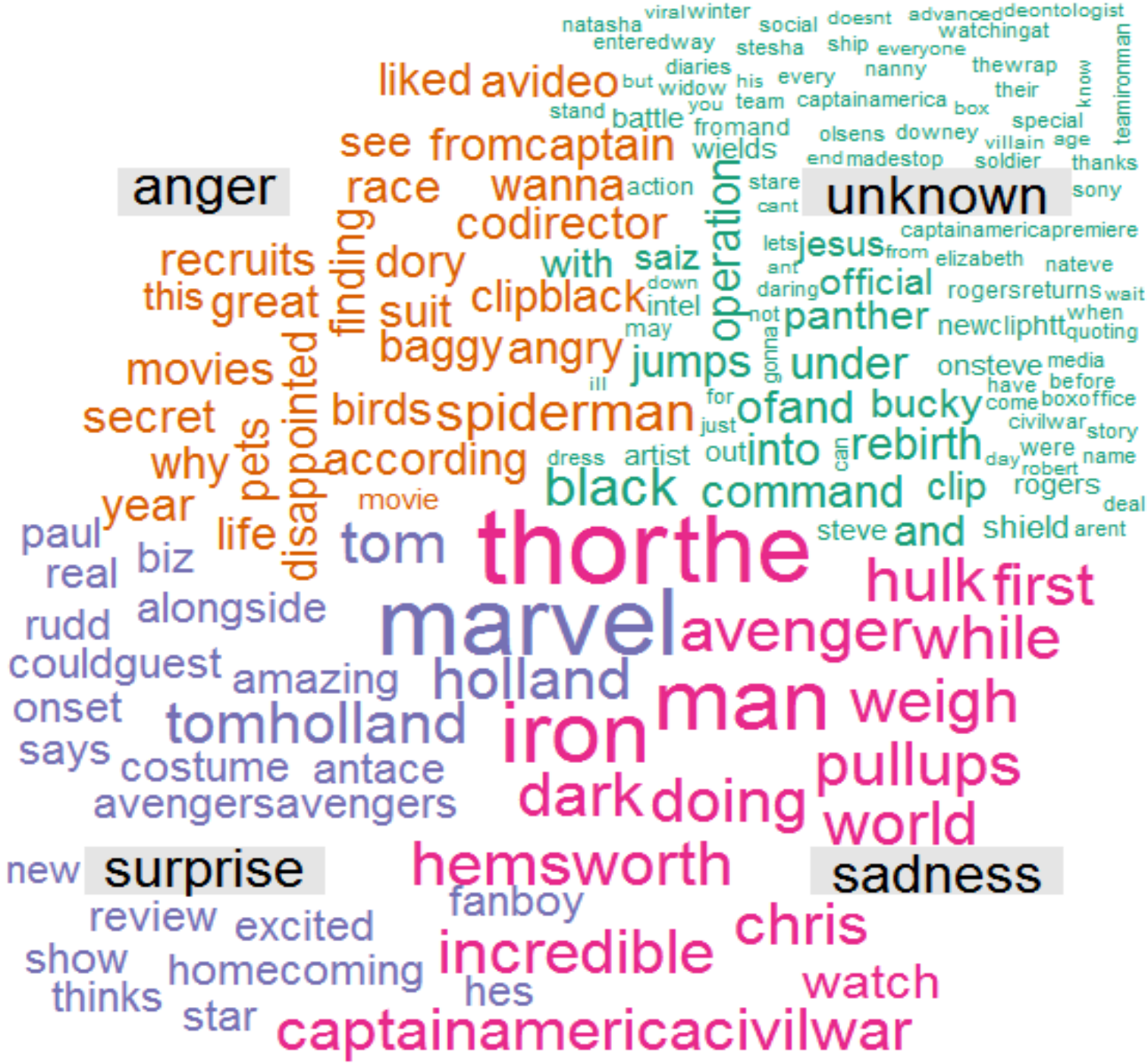
```
ggplot(sent_df, aes(x=emotion)) +
  geom_bar(aes(y=..count.., fill=emotion)) +
  scale_fill_brewer(palette="Dark2") +
  labs(x="emotion categories", y="number of tweets")
```



VISUALIZATION – Polarity Plot

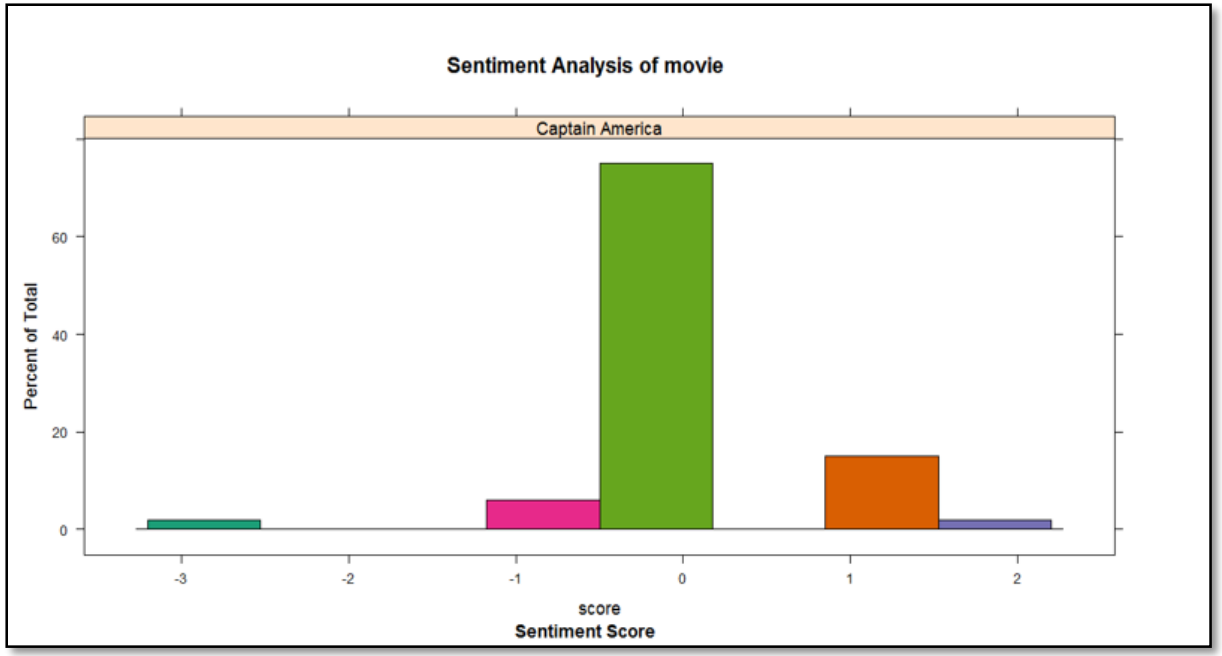
```
# plot distribution of polarity
ggplot(sent_df, aes(x=polarity)) +
  geom_bar(aes(y=..count.., fill=polarity)) +
  scale_fill_brewer(palette="RdGy") +
  labs(x="polarity categories", y="number of tweets") #+
# theme(title = "Sentiment Analysis of Tweets about Captain America\n(classification by polarity)",
# plot.title = theme_text(size=12))
})
```




anger
unknown
surprise
sadness
thorthe
marvel
man
iron
tomholland
holland
avenger
hulk
first
while
weigh
pullups
world
dark
doing
hemsworth
chris
incredible
watch
captainamericacivilwar
fanboy
review
excited
homecoming
hes
star
thinks
show
new
amazing
couldguest
onset
says
costume
antace
avengersavengers
alongside
rudd
real
biz
paul
year
life
tom
movie
pets
disappointed
secret
why
according
birds
spiderman
movies
baggy
angry
jumps
under
black
command
clip
rogers
and
steve
shield
rebirth
into
ofand
bucky
onsteve
panther
official
jesus
operation
codirector
wanna
race
see
fromcaptain
wields
liked
avideo
recruits
dory
finding
suit
clipblack
with
saiz
this
great
natasha
viralwinter
social
doesnt
advanced
deontologist
watchingat
enteredway
stesha
ship
everyone
diaries
thewrap
nanny
widow
his
every
but
their
team
captainamerica
box
stand
battle
fromand
special
olsens
downey
villain
age
end
madestop
soldier
thanks
stare
cant
sony
captainamericapremiere
lets
from
elizabeth
nateve
ant
daring
rogersreturns
wait
down
intel
not
when
may
gonna
quoting
media
before
have
come
boxoffice
civilwar
story
were
day
robert
name
deal
arent
for
just
dress
artist
out
can
ill
teamironman
know
you

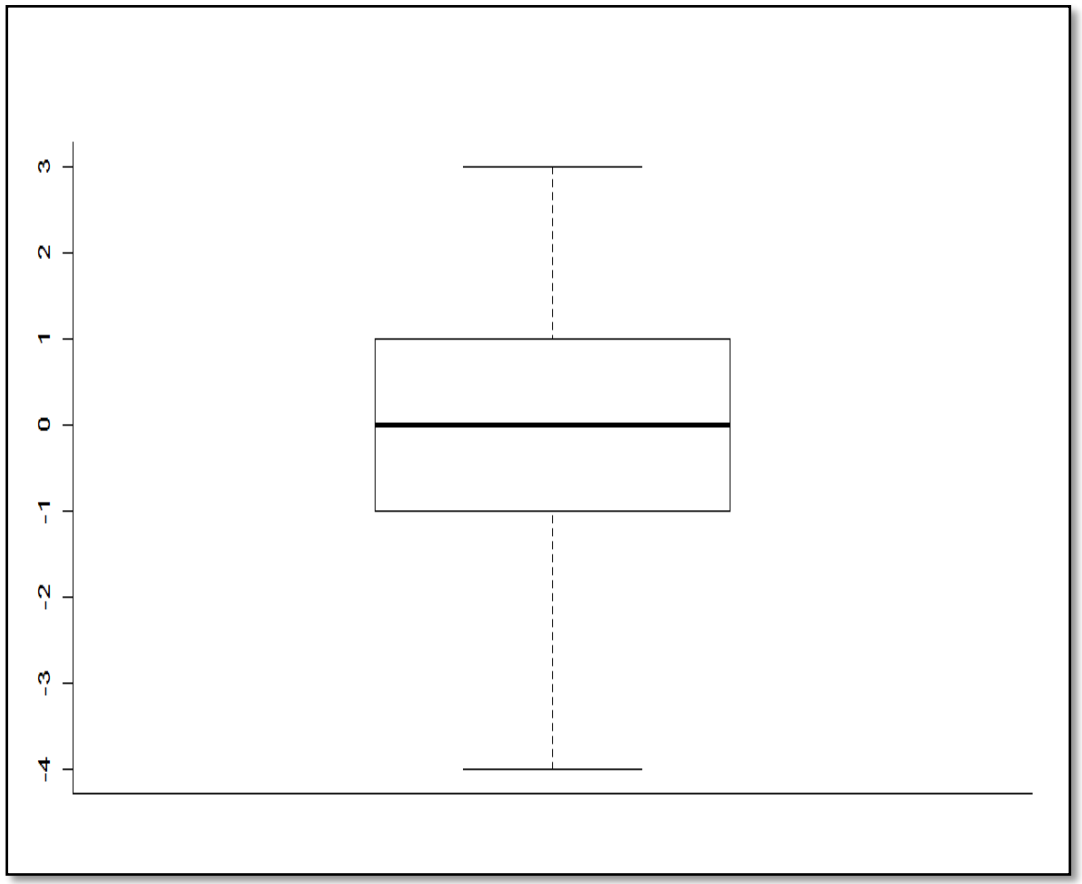
Sentiment Analysis based on scores for movie entered by user

```r
score.sentiment = function(sentences, pos.words, neg.words, .progress='none')
{

  scores = laply(sentences,
                 function(sentence, pos.words, neg.words)
                 {

                   # split sentence into words with str_split (stringr package)
                   word.list = str_split(sentence, "\\s+")
                   words = unlist(word.list)
                   # compare words to the dictionaries of positive & negative terms
                   # find the first occurrence of the first argument in the second argument:
                   pos.matches = match(words, pos.words)
                   neg.matches = match(words, neg.words)
                   # get the position of the matched term or NA
                   # we just want a TRUE/FALSE
                   pos.matches = !is.na(pos.matches)
                   neg.matches = !is.na(neg.matches)
                   # final score
                   score = sum(pos.matches) - sum(neg.matches)
                   return(score)
                 }, pos.words, neg.words, .progress=.progress )
  # data frame with scores for each sentence
  scores.df = data.frame(text=sentences, score=scores)
  return(scores.df)
}
```



BOX PLOT

```
    nooftweets = c(length(text))
    movie<-c(text)
    #applying function score.sentiment
    scores = score.sentiment(movie, pos, neg, .progress='text')
    scores$movie = factor(rep(c(input$select1), nooftweets))
    par(bty="l")
  # write.csv(scores,file="scores.csv")
    boxplot(score~movie, data=scores) #making a boxplot of sentiments
  })
}
```



Calculating mode to predict the number of movie screens

```
modescore<-mode(final$score)
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

mode<-getmode(final$score)
number<-function(x) {

  #if(x==-10 | x==-9 | x==-8) y<-100
  if(x==-7 | x==--6 | x==-5) y<-50
  if(x==-4 | x==-3 | x==-2) y<-100
  if(x==-1) y<-150
  if(x==0) y<-200
  if(x==1) y<-250
  if(x==2) y<-300
  if(x==3 | x==4 | x==5) y<-400
  if(x==6 | x==7 | x==8) y<-500
  return(y)
}
noOfScreens<-sapply(mode,number)
```

Using shiny server to integrate front end with the analysis performed at back end

Tweet Based Intelligent Movie Marketing

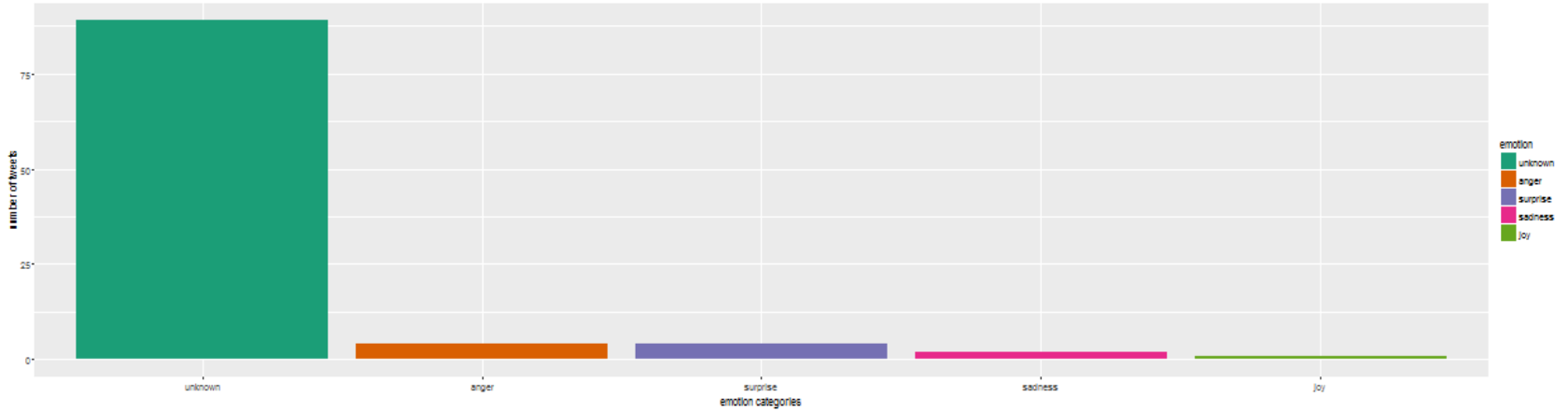
Select location

Boston

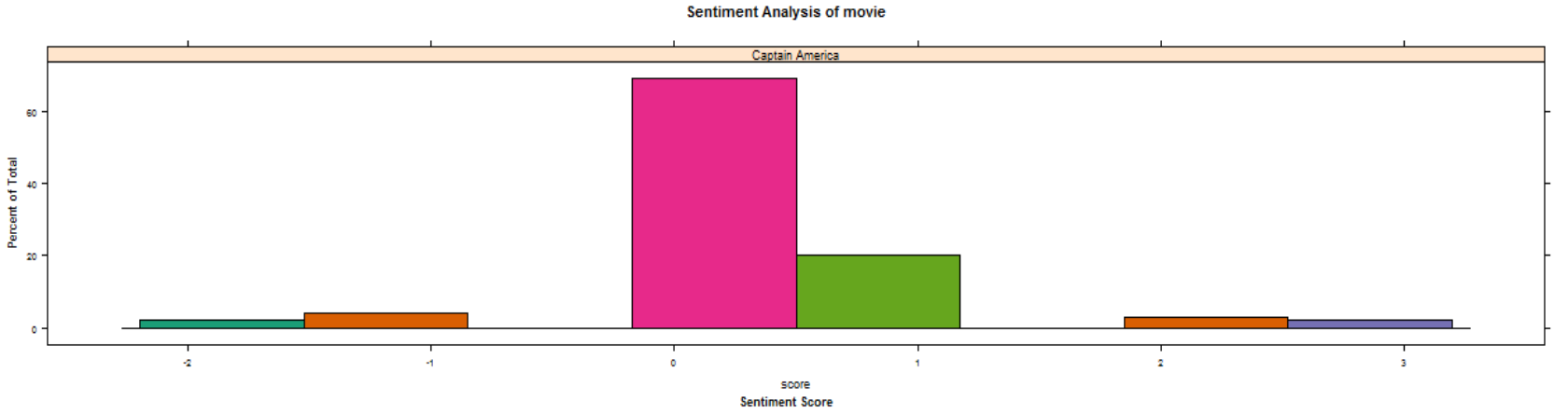
Select movie

Captain America

Plot of emotions | Plot of Polarity | Word Cloud | Suggested number of screens | Top Users



Histogram of Sentiment Scores | Boxplot of Sentiment Scores



User inputs city of his choice to view public opinion in that city

```
fluidRow(column(3, selectInput("select", label = h3("Select location"),
                        choices = c("Boston"="40.698470,-73.951442,50mi", "NewYork"="40.698470,-73.951442,50mi", "Los Angeles"="3
                        selected = 1)  ) )
```

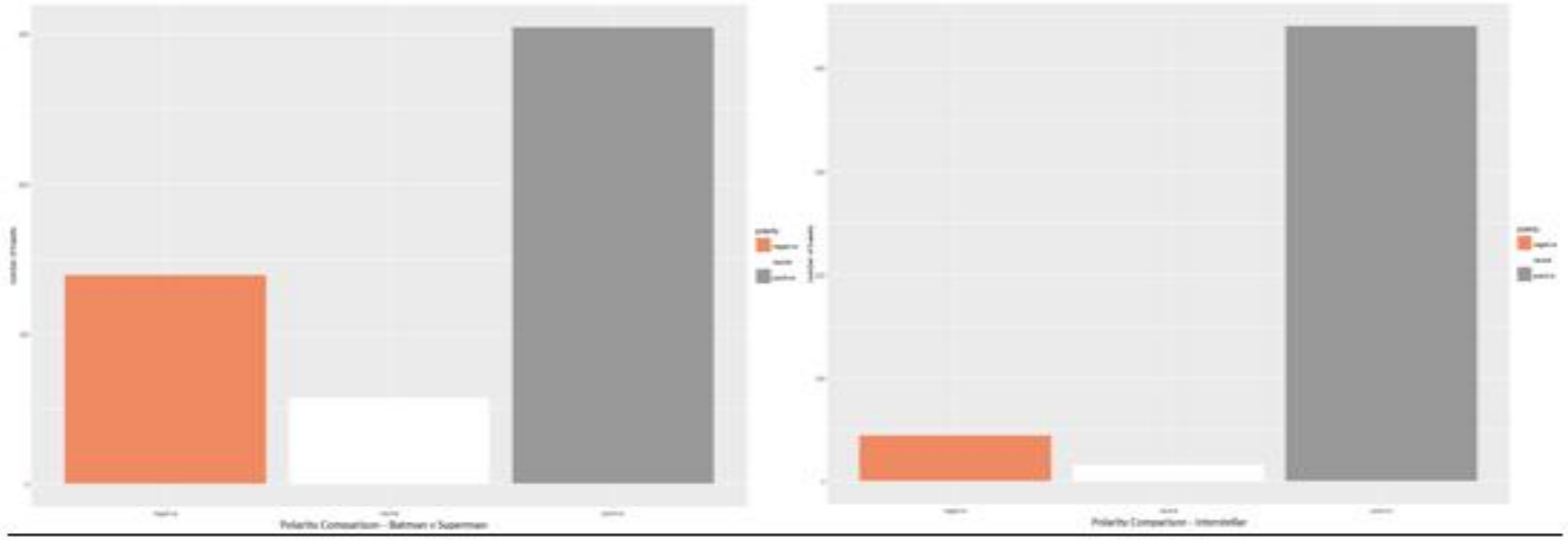
```
output$emotionplot<-renderPlot({
  captainamericatweets = searchTwitter("Captain America",n = 100, lang = "en",geocode=input$select)
  #head(captainamericatweets)
```

User inputs movie of his choice to view public opinion about that movie

```
,
fluidRow(column(3, selectInput("select1",label = h3("Select movie"),
                               choices = c("Captain America"="Captain America","X-Men"="X-Men"),selected=1))

)
```

```
output$histogramscores<-renderPlot({
  captainamericatweets = searchTwitter(input$select1,n = 100, lang = "en",geocode="40.712940,-73.987920,3000mi")
  #head(captainamericatweets)
```

Power of Sentiments



Conclusion

1. Learned how to integrate R and Shiny for text mining, sentiment analysis, visualization and deployment on the web. Using these tools together enables us to answer detailed questions.

2. It can be easily visualized that based on the sentiment analysis performed on the movie trailer the distributor can get a fair idea about the ideal location and number of screens to distribute the movie.

3. The opinion mining can be performed for any movie at any location, even for entire globe. By inputting any movie, a distributor can view the public opinion for a movie of his/her counterpart too. Based on that, he can make strategic decisions and do the needful to improve public feedback for his own movie.


Thank You!!!!