Facebook Data Mining and Sentiment Analysis Using R Language

Rajkumari Khobragade
M.tech Scholar(CSE)
PIET, Nagpur

Leena H. Patil
Prof. (CSE)
PIET, Nagpur

Abstract-Social media constitute a subject challenging source of new information for fetching and gathering and opinion making. Facebook is the most popular social media sites and often becomes the primary source of information. Facebook messages are short and well suited for knowledge discovery. Facebook provides both researchers and practitioners a free Application Programming Interface (API) which allows them to gather and analyze large data sets. Facebook information isn't solely opining texts, as facebook API provides a lot of info to perform attention- grabbing analysis studies. The paper concisely describes method of knowledge gathering and therefore the main areas of knowledge mining, information discovery and information visual image from facebook information. In this paper we can analysis a facebook data .we can fetch the Facebook text on a particular topic and stored it into R and then we can apply several text mining steps on the text to pre-process the text and than we can analyze the preprocess data by visualizing them.

Keywords: Facebook data, data mining, visualization, NLP, sentiment analysis, R language.

I. Introduction

Over past ten years, industries and organizations doesn't have demand to store and perform operations and analytics on info of the consumers. However around from 2005, the requirement to rework everything into information is way amused to satisfy the wants of the individuals. Therefore huge information came into image within the real time business analysis of process information. From twentieth century ahead this World Wide Web has modified the means of expressing their views. gift scenario is totally they're expressing their thoughts through on-line blogs, discussion forms and additionally some on-line applications like Facebook, Facebook, etc [3]. If we have a tendency to take Facebook as our example nearly 1TB of text information is generating inside per week within the sort of text. So, by this it's perceive clearly however this web is ever-changing the means of living and elegance of individuals. Among these text will be classified by the hash worth tags that they're commenting and posting their text. So, currently several corporations and additionally the survey corporations area unit mistreatment this for performing some analytics[5] specified they will predict the success rate of their product or additionally they will show the various read from the information that they need collected for analysis. But, to calculate their views is extremely troublesome during a traditional means by taking these serious information that area unit about to generate day by day.

Text mining has become a preferred approach to analysing and understanding massive datasets not done by the traditional analysis techniques. These tools are applied to a spread of data issues, like understanding themes in social media or facilitating Info retrieval in unstructured information. Text mining is also a extraordinarily great tool at intervals the beginnings of research exploration, allowing the matter info to counsel themes and concepts to the scientist throughout analysis. this may provides a useful begin line for framing further analysis queries and analysis approaches, notably if hypotheses Associate in Nursing any queries don't seem to be proverbial (as is typical with an inductive analysis approach). Moreover, these tools can also assist in improvement and structuring text-based info for future analysis in representation or completely different graphical tools. And, in addition to the tangible analysis benefits, text mining is also a fun and fruitful technique of discovery! Text Mining [4], is one of the foremost frequent however tough exercise faced by beginners in IP / analytics consultants. the most important challenge is one has to completely assess the underlying patterns in text, that too manually. For example: it's pretty common to delete numbers from the text before we have a tendency to do any reasonably text mining. However if we would like to extract one thing like what and were. Hence, the text cleansing exercise is very customized as per the target of the exercise and therefore the kind of text patterns.

R [9] is each a language and surroundings orienting towards applied math computing and graphics creation (R Core Team, 2016). R is created on the market below the antelope General Public License; as results of sturdy community involvement, there are various extensions, referred to as packages, developed over time, likewise as sturdy documentation. For this extensibility and flexibility, R has remained consistently common for information and text mining applications across many domains, and includes powerful text mining tools.

Here, we'll specialize in R packages helpful in understanding and extracting insights from the text and text mining packages.

This paper, we are going to be following subsequent packages:

1. tm, framework for text mining applications
2. SnowballC, text stemming library
3. ggplot2, one of the best data visualization libraries
4. Word cloud, for making word cloud visualizations

II. Literature Review

According to [1], Text mining, conjointly noted as text data processing, is that the method of extracting attention-grabbing and non-trivial patterns or data from text documents. It uses algorithms to rework free flow text (unstructured) into knowledge which will be analysed (structured) by applying applied mathematics, Machine Learning and language process (NLP) techniques. Text mining is associate degree evolving technology that enables enterprises to know their customers

Well, and facilitate them in redefining client desires. As e-commerce is changing into additional and skilful, the quantity of client reviews and feedback that a product receives has big quickly over a amount of your time. For a popular asset, the number of review comments can be in thousands or even more. This makes it difficult for the manufacturer to read all of them to make an informed decision in improving product quality and support. Again it is difficult for the manufacturer to keep track and to manage all customer opinions. This article attempts to derive some meaningful information from asset reviews which will be used in enhancing asset features from engineering point of view and helps in improving the support quality and customer experience.

In [8], they present a system for the acquisition, analysis and visualisation of Facebook data. Facebook messages are harvested and keep in a very distributed cluster, and also the knowledge is processed victimization algorithms enforced in a very MapReduce framework. we tend to gift a clump rule capable of characteristic the most topics of interest in a very tweet knowledge set. Also, we tend to design a visualisation technique that permits to follow the intensity of Facebook activity at a given geographical location. during this paper we've got bestowed a system for the acquisition, analysis and visualization of Facebook knowledge. Facebook messages are harvested and keep in a very distributed cluster, and he knowledge is processed victimization algorithms enforced in a very MapReduce framework. we tend to bestowed a clump rule capable of characteristic hot topics of interest in a very tweet knowledge set. Also, we tend to design a visualisation technique that permits to follow the density of Facebook activity in a very given geographical location. The system could be a model and was meant to gift the potential use of a social media platform as supply of enormous scale spatio-temporal data. It represents the building ground for future social media connected applications targeting a mess of doable applications with high social impact like emergency state of affairs management, risk and harm assessment and even social unrest.

In this paper they can visualize the Facebook data using matlab and matlab is a traditional technique which cannot handle bigdata, And Facebook data generates huge amount of data per data which is not able to process by traditional tools and technique, due to which we need a powerful visualizing techniques which can work on bigdata directly.

In [2], Facebook, as a social media could be a very fashionable manner of expressing opinions and interacting with others within the on-line world. Once taken in aggregation text will give a mirrored image of public sentiment towards events. During this paper, we offer a positive or negative sentiment on Facebook [12] posts employing a well-known we tend to use manually labeled (positive/negative) text to make a trained methodology to accomplish a task. The task is probing for a correlation between Facebook sentiment and events that have occurred. The trained model relies on the We tend to used external lexicons to notice subjective or objective text, other Unigram and written word options and used TF-IDF (Term Frequency-Inverse Document Frequency) to strain the options. Exploitation the FIFA journey 2014 as our case study, we tend to used Facebook Streaming API and a few of the official tourney hashtags to mine, filter and method text, so as to research the reflection of public sentiment towards sudden events. An equivalent approach will be used as a basis for predicting future events. Facebook, one in all the foremost

Common on-line social media and micro-blogging services could be a very fashionable methodology for expressing opinions and interacting with others within the on-line world. Facebook messages give real data within the format of short texts that categorical opinions, ideas and events captured within the moment. (Facebook posts) are well-suited sources of streaming information for opinion mining and sentiment polarity detection Opinions, evaluations, emotions and speculations usually mirror the states of individuals; they contains narrow-minded information expressed during a language composed of subjective expressions .During this paper, we tend to examine the effectiveness of a usually used text categorization methodology known as Bayesian supply Regression (BLR) Classification for providing positive or negative sentiment on text. We tend to use extracted Facebook sentiment to seem for correlations between this sentiment and major FIFA journey 2014 events as our case study.

This paper the calculated the polarity of the text with the help of Bayesian supply Regression classification methodology and predict some events based on correlation between the events. But this methodology fails when data is very huge in terms of pettabyte and also it cannot do real time analysis, for this we need a new tool and technique which can handle such huge and large datasets.

III. Problem Definition

Text mining [7] facilitate a company derive doubtless valuable business insights from text-based content like word documents, email and postings on social media streams like Facebook, Facebook and LinkedIn. data processing or data mining plays a absolutely role in deciding as a result of through these mining techniques we will analyse the info and on the idea of result we will take a call. Now a days social media sites like Facebook are widely used to share user opinions on various topics, Facebook gives a platform to user to share their views and thoughts on various field like political, industrial, education and there is a petabytes of data generated by Facebook in a day.

So the mining techniques are used to analysis the social Facebook data thorough we get large amount of datasets to analysis, so the analysis of Facebook data provides a better way for making decision.

A. Motivation

Text mining [11] is not yet a part of mainstream predictive analytics, though it is on the short list for many organizations. But text mining is difficult, requiring additional expertise and processing complementary to predictive modelling, but not often taught in machine learning or statistics computing.

Nonetheless, data mining is being used increasingly as organizations recognize the untapped information contained in text. Social media, such as Facebook [13] and Facebook, have been used effectively by organizations to uncover trends that, when identified through text mining, can be used to leverage the positive trends.

Data preprocessing could be a data processing technique that involves reworking information into a visible format. Real- world information is commonly incomplete, inconsistent, and/or lacking in sure behaviours or trends, and is probably going to contain several errors. Information preprocessing could be a well-tried methodology of resolution such problems. Information preprocessing prepares information for any process.

Preprocessing steps contains many objectives. They're as follows:

1. Fetch information from Facebook Streaming API.
2. Convert unstructured JSON information into structured information.
3. Apply stemming to get rid of stop words.
4. Store the preprocessed information for analysis.

IV. Proposed Work

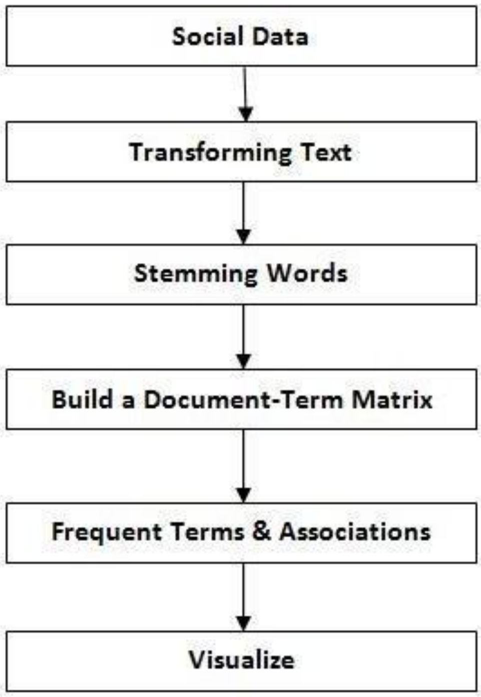
The work involved in this usually requires several computational techniques (such as data and text mining, natural language processing, etc.) and complex analytical processes required to manipulate varied data sources. Besides that, reach a point of balance between the computational side of the process and the aesthetic side using tables, charts, colors and other visual features, could favor a good analysis and quicker understanding of such data. In a process to derive the target outcome from the unstructured raw text which we fetching from web, the first step is to identify suitable data source.

Pre-processing is a first step plays a very important role in text mining techniques and applications. It is the method in the text mining process. In this dissertation, we discuss the three packages comes in R language through which we can perform the text mining on Facebook data. Text mining of Facebook data with R packages Facebook, tm and ggplot2

V. Proposed Methodology

Algorithm Steps will follow:

1. First we get a complex social data and stored.
2. when retrieving we have a tendency to reworked the text, After that, the corpus desires a handful of transformations, as well as dynamical letters to small letter, removing punctuations/numbers and removing stop words.
3. In several cases, words got to be stemmed to retrieve their radicals. as an example, "example" and "examples" area unit each stemmed to "exampl". However, after that, one may want to complete the stems [6] to their original forms, so that the words would look "normal".
4. After changing and stemming process is done then we build a document term matrix. Based on the confusion matrix, many text mining tasks can be done, for example, clustering, classification and association analysis.
5. With the help of matrix we can identify the frequent words and their association between words.
6. After building a document-term matrix, we can now visualize the outputs.

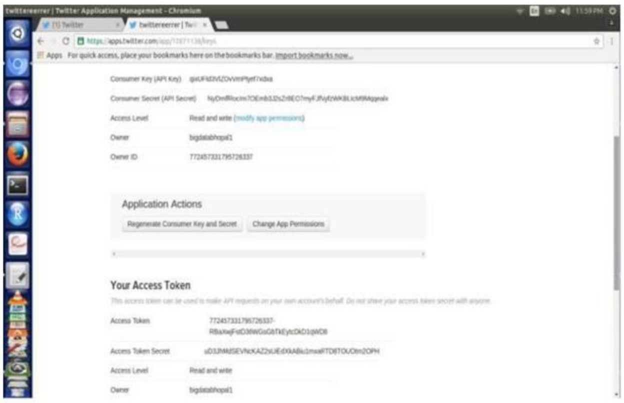


Proposed Flow Diagram

B.Gathering the data

Any social media investigation is only as good as the data used for its analysis. The process of social media analysis involves essentially four steps: data identification, data analysis, data interpretation and, finally, information presentation. The main problem is how to extract the information that is available on Facebook and how it can be used to draw meaningful insight. To achieve this, first there is a need to build a data analyser for text data.Text data are available to researchers and practitioners through public Facebook APIs. Facebook allows developers to collect data via Facebook REST API. (https://dev.Facebook.com/rest/public/) and the Streaming API (https://dev.Facebook.com/streaming/overview).

First of all if we want to do analysis on Facebook data we want to get Facebook data first so to get it we want to create an account in Facebook developer and create an application by clicking on the new application button provided by them shown in figure 1 , After creating a new application just create the access tokens so that we need to provide our authentication details there and also after creating application it will be having one consumer keys to access that application for getting Facebook data.


twittereerrer | Twitter Application Management - Chromium
(5) Twitter
twittereerrer | Tw
https://apps.twitter.com/app/13871138/keys
Apps For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Regenerate Consumer Key and Secret
Change App Permissions

Consumer Key (API Key) qixUFld3VlZOvVmPlyef7xdxa
Consumer Secret (API Secret) NyDmf8ocim7OEmb3JzsZr8EO7myFJfvyfzWKBLIcM9Aggeals
Access Level Read and write (modify app permissions)
Owner bigdatabhopal1
Owner ID 772457331795726337

Application Actions

Your Access Token

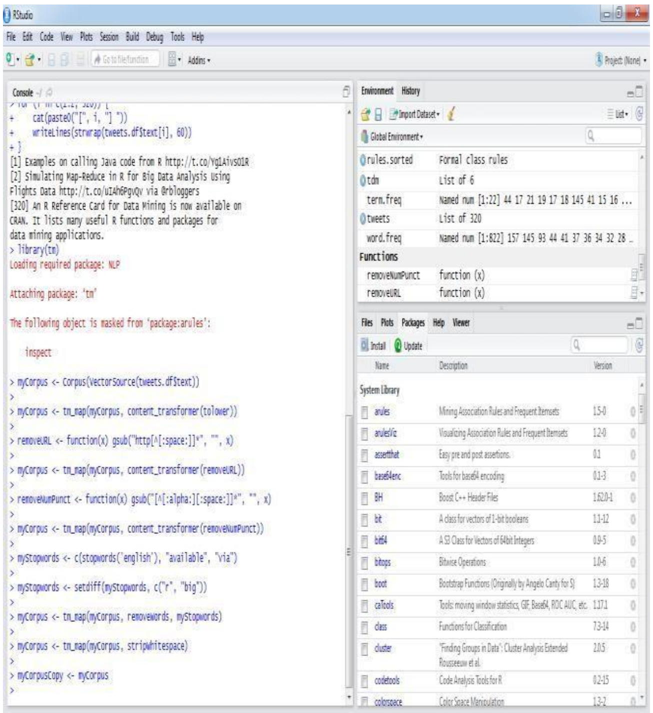
This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.

Access Token 772457331795726337-RBaXwjFstD36WGsGbTkEytcDkD1qWD8
Access Token Secret sD3JhMdSEVNcKAZ2sUEdXkABu1mxaRTD8TOUOtn2OPH
Access Level Read and write
Owner bigdatabhopal1

Figure 1 Access keys from Facebook application

C. Perform text mining on facebook

For performing text mining on text we need a package called tm package which internally consist natural language processing NLP package. Figure 2 shows the loading and performing operation through tm package.



```
> for (i in c(1:2, 320)) {
+     cat(paste0("[", i, "] "))
+     writeLines(strwrap(tweets.df$text[i], 60))
+ }
[1] Examples on calling Java code from R http://t.co/Yg1AivsO1R
[2] Simulating Map-Reduce in R for Big Data Analysis Using
Flights Data http://t.co/uIAh6PgvQv via @rbloggers
[320] An R Reference Card for Data Mining is now available on
CRAN. It lists many useful R functions and packages for
data mining applications.
> library(tm)
Loading required package: NLP

Attaching package: 'tm'

The following object is masked from 'package:arules':

    inspect

> myCorpus <- Corpus(VectorSource(tweets.df$text))
>
> myCorpus <- tm_map(myCorpus, content_transformer(tolower))
>
> removeURL <- function(x) gsub("http[^[:space:]]*", "", x)
>
> myCorpus <- tm_map(myCorpus, content_transformer(removeURL))
>
> removeNumPunct <- function(x) gsub("[^[:alpha:][:space:]]*", "", x)
>
> myCorpus <- tm_map(myCorpus, content_transformer(removeNumPunct))
>
> myStopwords <- c(stopwords('english'), "available", "via")
>
> myStopwords <- setdiff(myStopwords, c("r", "big"))
>
> myCorpus <- tm_map(myCorpus, removeWords, myStopwords)
>
> myCorpus <- tm_map(myCorpus, stripWhitespace)
>
> myCorpusCopy <- myCorpus
>
```

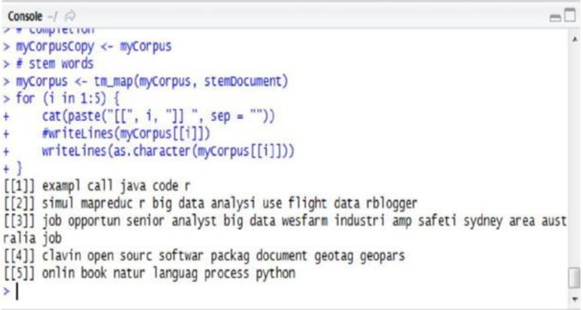
```
rules.sorted    Formal class rules
tdm             List of 6
term.freq       Named num [1:22] 44 17 21 19 17 18 145 41 15 16 ...
tweets          List of 320
word.freq       Named num [1:822] 157 145 93 44 41 37 36 34 32 28 ...
Functions
removeNumPunct  function (x)
removeURL       function (x)
```

Name	Description	Version
System Library		
arules	Mining Association Rules and Frequent Itemsets	1.5-0
arulesViz	Visualizing Association Rules and Frequent Itemsets	1.2-0
assertthat	Easy pre and post assertions.	0.1
base64enc	Tools for base64 encoding	0.1-3
BH	Boost C++ Header Files	1.62.0-1
bit	A class for vectors of 1-bit booleans	1.1-12
bit64	A S3 Class for Vectors of 64bit Integers	0.9-5
bitops	Bitwise Operations	1.0-6
boot	Bootstrap Functions (Originally by Angelo Canty for S)	1.3-18
caTools	Tools: moving window statistics, GIF, Base64, ROC AUC, etc.	1.17.1
class	Functions for Classification	7.3-14
cluster	"Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.	2.0.5
codetools	Code Analysis Tools for R	0.2-15
colorspace	Color Space Manipulation	1.3-2

Figure 2 Text preprocessing using tm packages

We are using corpus for preprocessing Facebook text and tm package consist many function through which we can remove numbers, punctual, and special characters coming from text. And we are converting first all the text data into lower case to remove noise and we can also eliminate url's coming from text.

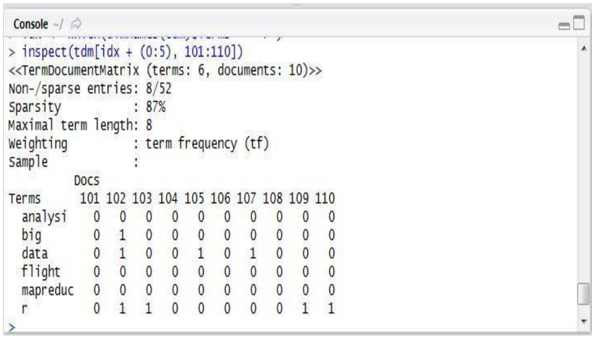
After preprocessing using tm package we can perfrom stemming on preprocessed data for this we can use SnowballC package. Figure 3 describe the stemming is perform on preprocesssed text. For this we can collects same meaning types word separately such as example, exampl, examples comes into same category.



```
> # completion
> myCorpusCopy <- myCorpus
> # stem words
> myCorpus <- tm_map(myCorpus, stemDocument)
> for (i in 1:5) {
+     cat(paste("[[", i, "]] ", sep = ""))
+     #writeLines(myCorpus[[i]])
+     writeLines(as.character(myCorpus[[i]]))
+ }
[[1]] exampl call java code r
[[2]] simul mapreduc r big data analysi use flight data rblogger
[[3]] job opportun senior analyst big data wesfarm industri amp safeti sydney area aust
ralia job
[[4]] clavin open sourc softwar packag document geotag geopars
[[5]] onlin book natur languag process python
> |
```

Figure 3 Stemming on preprocessed text

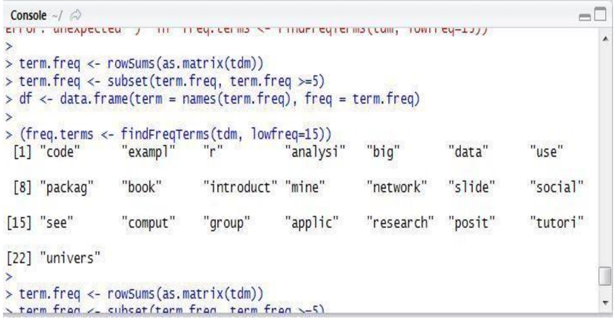
After stemming process we can develop a term-document matrix and find the frequency of each term using association properties. Figure 4 describe the operation on tdm (term document matrix) and finding most frequent keywords.



```
> inspect(tdm[idx + (0:5), 101:110])
<<TermDocumentMatrix (terms: 6, documents: 10)>>
Non-/sparse entries: 8/52
Sparsity           : 87%
Maximal term length: 8
Weighting          : term frequency (tf)
Sample             :
          Docs
Terms      101 102 103 104 105 106 107 108 109 110
  analysi    0   0   0   0   0   0   0   0   0   0
  big        0   1   0   0   0   0   0   0   0   0
  data       0   1   0   0   1   0   1   0   0   0
  flight     0   0   0   0   0   0   0   0   0   0
  mapreduc   0   0   0   0   0   0   0   0   0   0
  r          0   1   1   0   0   0   0   0   1   1
>
```

Figure 4 Document matrix

After creating document term matrix we can find the frequent terms and find those terms that are having maximum frequency. Figure 5 shows the frequent terms.



```
Error: unexpected ')' in "freq.terms <- findFreqTerms(tdm, lowfreq=15))"
>
> term.freq <- rowSums(as.matrix(tdm))
> term.freq <- subset(term.freq, term.freq >=5)
> df <- data.frame(term = names(term.freq), freq = term.freq)
>
> (freq.terms <- findFreqTerms(tdm, lowfreq=15))
 [1] "code"      "exampl"    "r"         "analysi"   "big"       "data"      "use"

 [8] "packag"    "book"      "introduct" "mine"      "network"   "slide"     "social"

[15] "see"       "comput"    "group"     "applic"    "research"  "posit"     "tutori"

[22] "univers"
>
> term.freq <- rowSums(as.matrix(tdm))
> term.freq <- subset(term.freq, term.freq >=5)
```

Figure 5 Frequent terms

D. Analyze the pre-process data

After text mining using tm package we can analyse the text mining result using visualizing package called ggplot2, In tm package we can find the frenquent terms and using ggplot2 we can visualize these frequent term. The ggplot for frequent terms along according to their count.

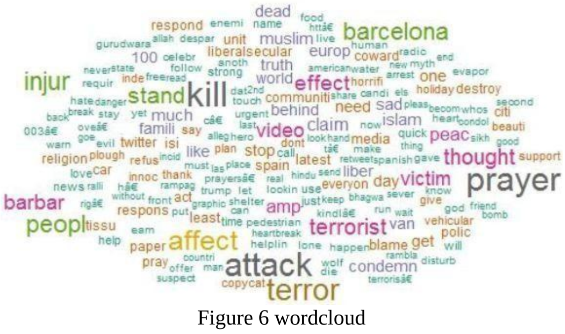


Figure 6 wordcloud

The word cloud in the figure shows that most frequently used words in the tweets are kill, prayer, attack, terror, affect, people and so on. The different colors and size of the words indicate their frequency for example kill, prayer and attack have higher frequency than other words. These words represent the immediate response and reaction of the people.[3]

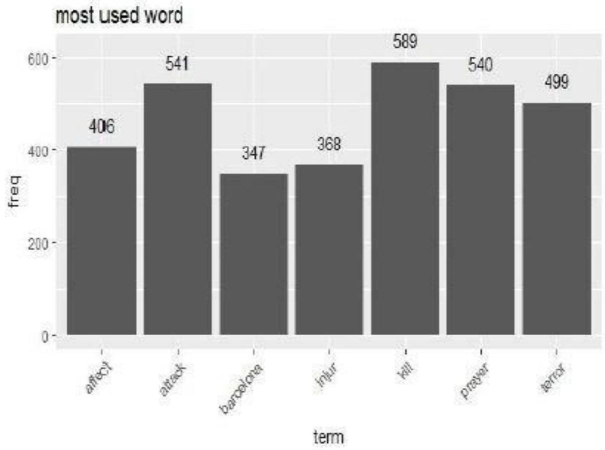


Figure 7 Most used words in the facebook

Fig 7 gives the visualization of the most used words in the text. It can be seen that the word kill has the highest frequency followed by words attack and prayer. It's quite evident as it was a terrorist attack and these words were expected to occur more frequently. Words like affect, injury and terror occur quite frequently too. [8]

VI. Sentimental Analysis Modes

R is an open source Programming language and software basically used by statisticians and data miners to study various statistical data's such as poles surveys etc. The complete procedure for analyzing sentiments has been detailed in several series of steps which is described below:

Step 1: Create your account on Facebook Application using Facebook Developers and login to gain access to tweets from R Studio. Download the access "token key" and "api key" to perform handshake with R console.

Step 2: Install and load the packages required for sentimental analysis. Some of the packages used are:
Facebook provides interface to the web API.

- Facebook httr – controls request and works with URLs.
- ROAuth – used to authenticate to the server of choice.
- ggplot2 – for visualization.
- Word cloud – to form word clouds and to visualize it stringr – to use wrappers.
- syuzhet – consist of sentiment dictionary and to extract sentiments.
- tm – used for textmining.

Step 3: Once the authentication is successfully done text are extracted using hash tag.

Step 4: This step involves text mining that is data cleaning, as already mentioned, tweets on Facebook contain many unwanted information, so it is important to clean the text and derive only the useful information.

Step 5: The cleaned data is arranged in data frame and matrix so that desired operation can be performed.

Step 6: The emotions are extracted from the tokenized words and visualization of analysis is done to observe the sentiments.

VII. Data Preparation AndModeling

The data gathered is not pure. It contains hash tags, urls, abbreviations, punctuation, stop words, etc. The data is needed to be cleaned to perform better analysis. The tm and stringr packages are to be used from the library of R for performing text mining. The data and text are formed into corpus, and then the corpus is invoked. The corpus is cleaned in proper order which is links, @, punctuations and other symbols which don't express any emotions. Corpus is used to carryout further text mining. [4][5]
Various functions are used to remove unwanted strings from the facebook text. [7]

- Remove Punctuation () is used to eliminate punctuation marks from tweets.
- Remove Numbers () is used to eliminate numbers as they don't have any underlying sentiment.
- tolower() converts the entire corpus content into lower case.
- Stopwords removes English words without sentiments like articles, conjunction etc.
- removewords is used to eliminate some specific words.
- stemDocument method reduces a word to its original root word. For example, the word "running" will be changed to its root form run.
- stripWhitespace is used to eliminate extra white spaces.

After performing data cleaning, the data is transformed into required format. Now, a cleaned corpus is transformed into document term matrix. A document terms matrix represent frequency of every word present in the corpus. [5]

VIII. Conclusion

Facebook data is very useful in decision making because its provide several of opinions on various topics. So the texting mining will perform on Facebook data and we are using visualizing techniques. In this paper we can analysis Facebook data from, we can fetch the Facebook data on a particular topic and stored it into R and than we can apply several text mining steps on the facebook to pre-process the facebook data and then we can analyze the preprocess data.

References

[1] Chandrasekhar Rangu, Shuvojit Chatterjee, Srinivasa Rao Valluru, "Text Mining Approach for Product Quality Enhancement" in IEEE 2017.

[2] Mr. Peiman Barnaghi and John G. Breslin , ", Opinion Mining and sentiment polarity on Twitter and correlation between Events & Sentiment", International Conference on Big Data Computing and Application, IEEE 2016

[3] Judith Sherin Tilsha S, Shobha M.S.," A Survey on Twitter Data Analysis Techniques to Extract Public Opinion." IJARCSE, Vol. 5, Issue 11, Nov 2015, 2277128X

[4] LokmanyathilakGovindanSankarSelvan,"A Framework for Fast-Feedback Opinion Mining on Twitter Data Streams", IEEE 2015

[5] T. K. Das, D.P. Acharjya & M. R. Patra, "Opinion Mining about a product by Analyzing Public Text in Facebook ", ICCCI- 2014, Jan 03-05, 2014.

[6] Porter M.F, Snowball: A language for stemming algorithms. 2001.

[7] Ning Zhong, Yuefeng Li, and Sheng-Tang Wu, "Effective Pattern Discovery for Text Mining", IEEE Transactions on Knowledge and Data Engineering, Vol. 24, No.1, January 2012.

[8] Andrei Sechelea, Tien Do Huu, Evangelos Zimos, and Nikos Deligiannis, "Twitter Data Clustering and Visualization", in 2016 23rd International Conference on Telecommunications (ICT), 2016 IEEE.

[9] Shruti Kohli, Himani Singal, "Data Analysis with R" in 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing.

[10] Arun Jalanila, Nirmal Subramanian, "Comparing SAS® Text Miner, Python, R" in 2016 IEEE International Conference on Healthcare Informatics