**SENTIMENT ANALYSIS OF WHATSAPP CHAT USING R**

**REPORT**

**REVIEW - 2**

**Submitted By**

**RA1511008010644 Shubham Singh RA1511008010620 Shilpa Suryawanshi**

**RA1511008010654 Sidhant Iyer**

**In partial fulfillment for award of degree**

**Of**

**BACHELOR OF TECHNOLOGY**

IN

INFORMATION TECHNOLOGY



**SENTIMENT ANALYSIS OF A WHATSAPP CHAT USING R**

**Abstract**

Data, the one, that made a trend and evolution from good old techniques to modern trendy techniques. Based on the source the data is extracted, its form differs that is in recent decades data are scalable, skeptical and diverse compared to old structured data that are derived from specific known sources.The next criterion for the technological evolution is, the storage space that is needed for data. Structured data storage lies in the database whereas for the unstructured and semi-structured ones the storage trend tips to cloud. The vital thing in addition to storage is data handling and analysis, both process when involved in trendy data results in a complexity in terms of containment, processing and visualization of data. To overcome these complexities on trendy unstructured data there emerged a concept called Big Data Analytics. It uses numerous tools and techniques that resolved the problem from data storage to visualization. One such technique is Analytical sandbox, the most central container that stores and handles data in a very appropriate manner. In addition, Big data tools provide a variety of analytical and visualization techniques that produces efficient graphical view of modern data. This paper centered on sentimental analysis of WhatsApp group data in which text mining was incorporated to the chat file and the resulted chat text file is further processed with analytical tools that analyze the chat contents and produces an graphical visualization of the sentiments shared in the group. Its outcome view lies in high echelon of positive opinions compared to other emotions such as anger, fear, disgust, anticipation, joy, sadness, surprise, trust and negative.

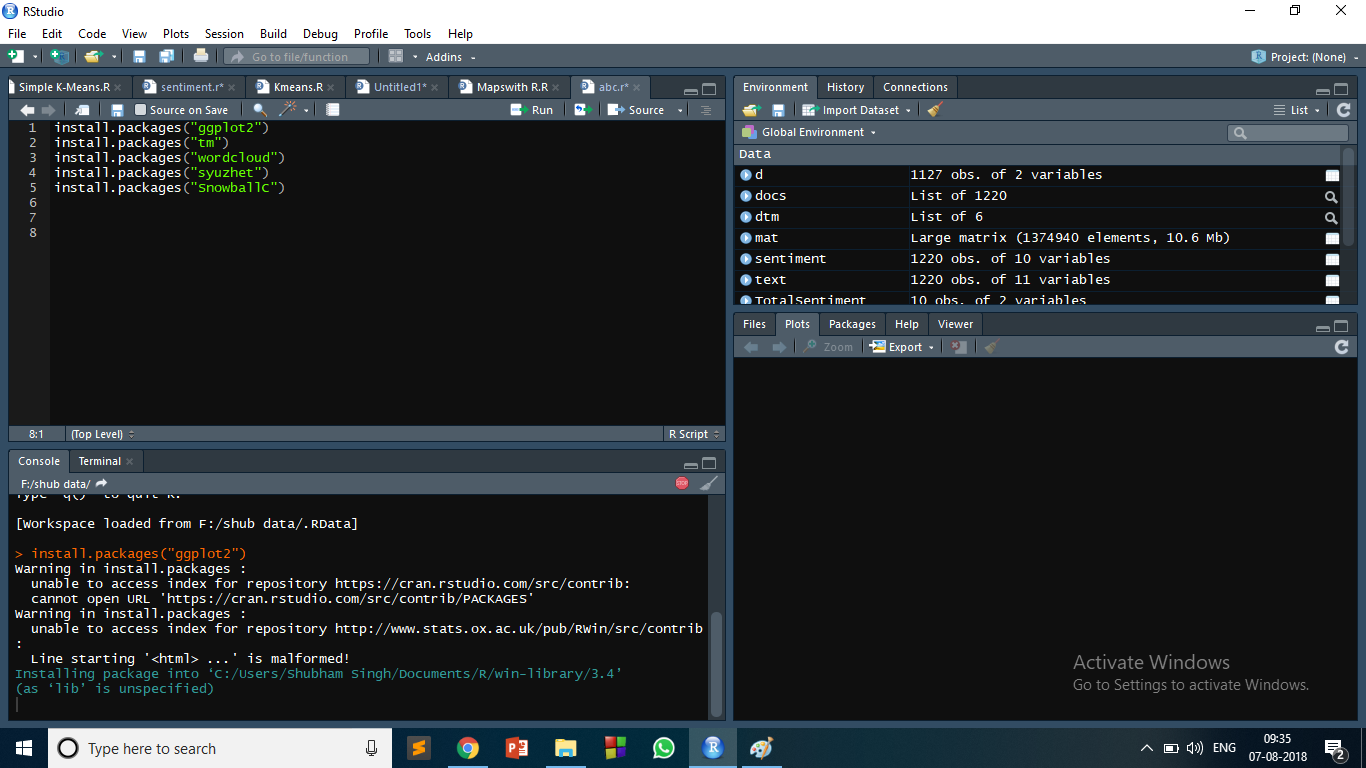
**Requirement analysis**

* **An Intel-compatible platform running Windows 2000, XP/2003/Vista/7/8/2012 Server/8.1/10**
* **2 GB RAM**
* **2 GB Memory**
* **R Studio 3.4.3**
* **R i386 3.4.3**
* **WhatsApp chat ( text format)**
* **Internet connection.**

**LEVEL 1 IMPLEMENTATION**

**Libraries Required :**

* **library(ggplot2)**
* **library(tm)**
* **library(wordcloud)**
* **library(syuzhet)**
* **library(SnowballC)**

****

1. **Ggplot2 :**

**The ggplot2 package, created by Hadley Wickham, offers a powerful graphics language for creating elegant and complex plots. Its popularity in the R community has exploded in recent years. There is a helper function called qplot() (for quick plot) that can hide much of this complexity when creating standard graphs.**

1. **TM – text mining**

**the tm\_map() function is used to remove unnecessary white space, to convert the text to lower case, to remove common stopwords like ‘the’, “we”.**

**The information value of ‘stopwords’ is near zero due to the fact that they are so common in a language. Removing this kind of words is useful before further analyses. For ‘stopwords’, supported languages are danish, dutch, english, finnish, french, german, hungarian, italian, norwegian, portuguese, russian, spanish and swedish. Language names are case sensitive.**

**You could also remove numbers and punctuation with removeNumbers and removePunctuation arguments.**

1. **Wordcloud**

**One can create a word cloud, also referred as text cloud or tag cloud, which is a visual representation of text data. The text mining package (tm) and the word cloud generator package (wordcloud) are available in R for helping us to analyse texts and to quickly visualize the keywords as a word cloud.**

1. **SnowballC**

SnowballC package is used for text stemming. Text stemming reduces words to their root form. In other words, this process removes suffixes from words to make it simple and to get the common origin. For example, a stemming process reduces the words “moving”, “moved” and “movement” to the root word, “move”.

1. SYUZHET

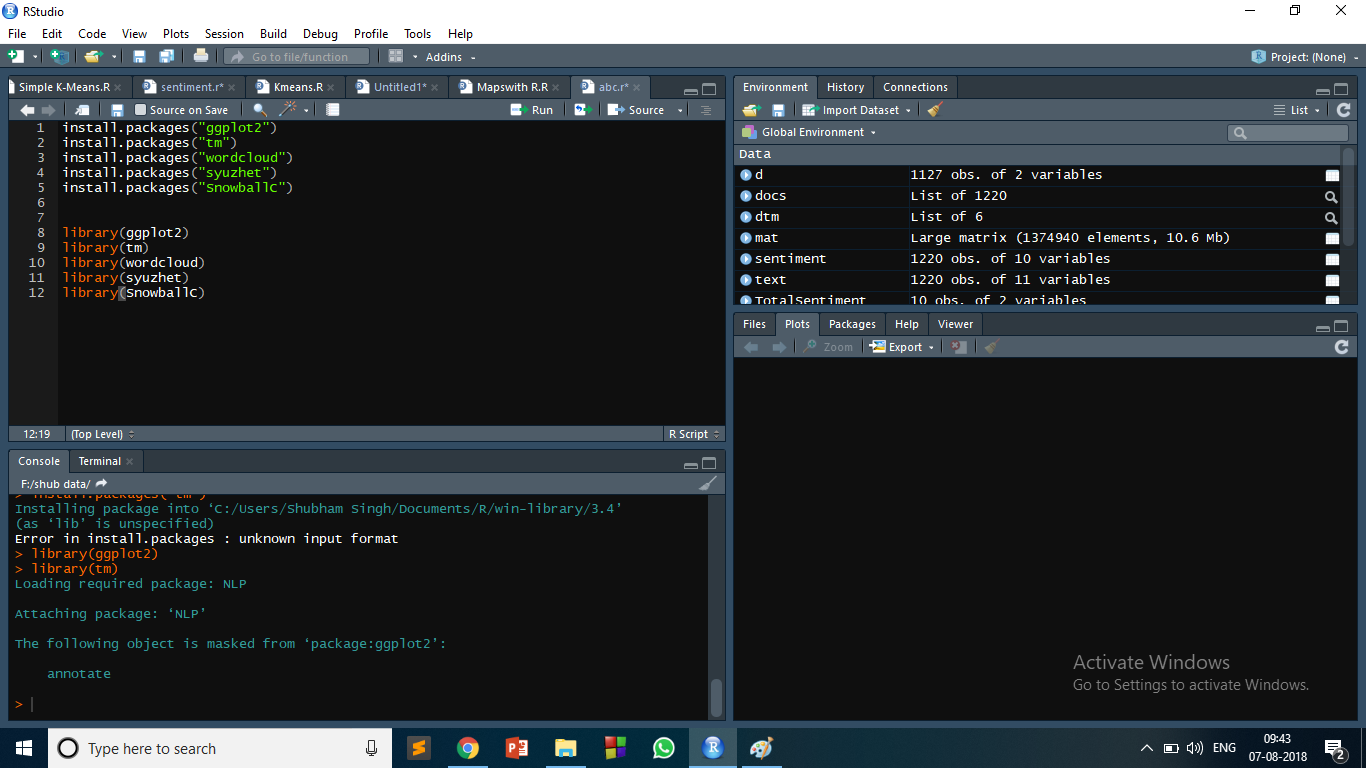
This vignette demonstrates use of the basic functions of the Syuzhet package. The package comes with four sentiment dictionaries and provides a method for accessing the robust, but computationally expensive, sentiment extraction tool. The goal of this vignette is to introduce the main functions in the package so that you can quickly extract plot and sentiment data from your own text files.

get\_nrc\_sentiment

The get\_nrc\_sentiment function returns a data frame in which each row represents a sentence from the original file. The columns include one for each emotion type was well as the positive or negative sentiment valence. .

**Loading the libraries**

* **library(ggplot2)**
* **library(tm)**
* **library(wordcloud)**
* **library(syuzhet)**
* **library(SnowballC)**

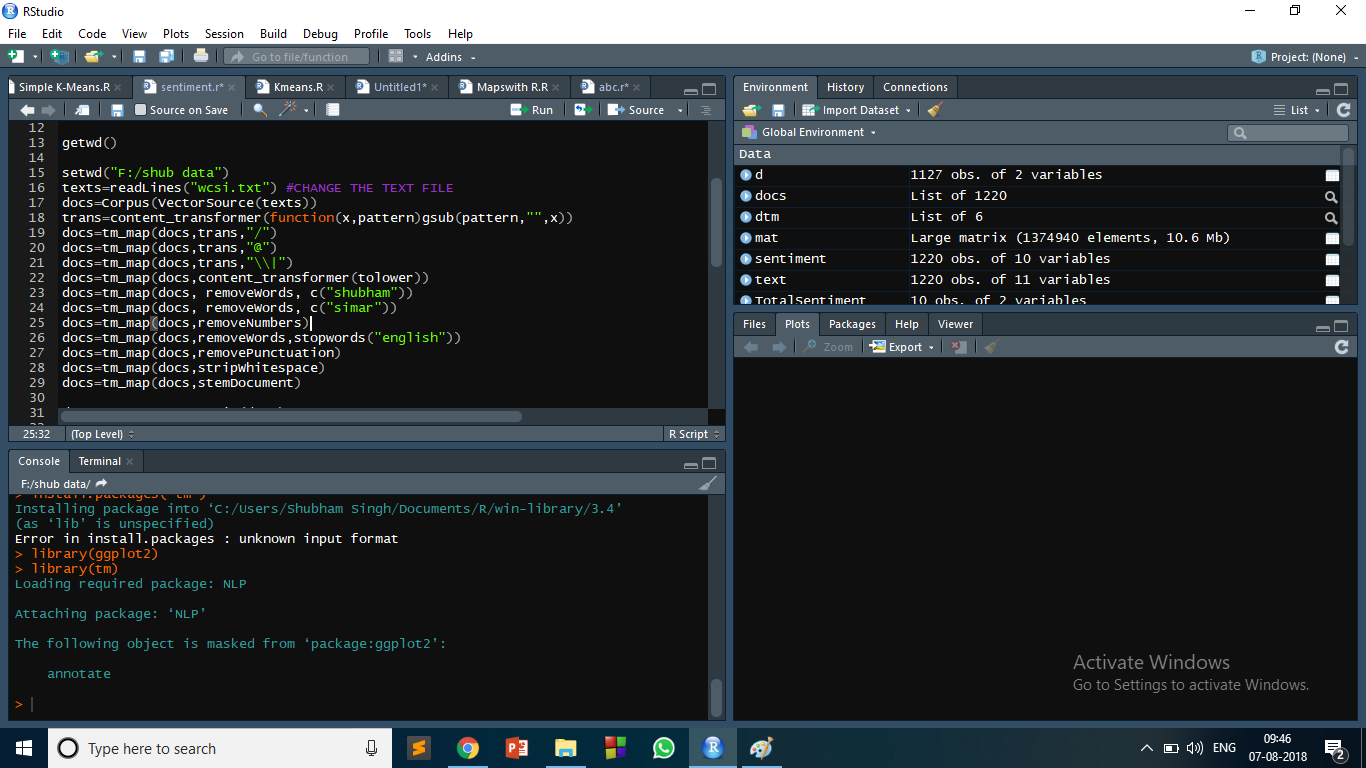
****

**Loading the text file**

* **setwd("F:/shub data")** //used to set the working directory.
* **texts=readLines("wcsi.txt**") //Reads the lines of the whatsapp chats and saves in the variable named as texts.
* **docs=Corpus(VectorSource(texts))** // It makes a collection of data that has to be operated**.**

**Cleaning the data**

* **trans=content\_transformer(function(x,pattern)gsub(pattern,"",x))**
* **docs=tm\_map(docs,trans,"/")**
* **docs=tm\_map(docs,trans,"@")**
* **docs=tm\_map(docs,trans,"\\|")**
* **docs=tm\_map(docs,content\_transformer(tolower))**
* **docs=tm\_map(docs, removeWords, c("shubham"))**
* **docs=tm\_map(docs, removeWords, c("simar"))**
* **docs=tm\_map(docs,removeNumbers)**
* **docs=tm\_map(docs,removeWords,stopwords("english"))**
* **docs=tm\_map(docs,removePunctuation)**
* **docs=tm\_map(docs,stripWhitespace)**
* **docs=tm\_map(docs,stemDocument)**

****