TABLE OF CONTENTS

[**CERTIFICATE 3**](#_Toc510464107)

[**INTRODUCTION: 6**](#_Toc510464108)

[**ABSTRACT: 6**](#_Toc510464109)

[**Problem Statement: 6**](#_Toc510464110)

[**LITERATURE REVIEW 7**](#_Toc510464111)

[***Social Media Analysis Techniques:* 12**](#_Toc510464112)

[***Social Media Analytics Tools:* 12**](#_Toc510464113)

[**Architecture / Methodology / Algorithm / Framework used: 13**](#_Toc510464114)

[**ALGORITHM: 11**](#_Toc510464115)

[**FRAMEWORK USED: 14**](#_Toc510464117)

[**LIBRARIES USED 15**](#_Toc510464118)

[***TextBlob* 15**](#_Toc510464119)

[***CSV* 15**](#_Toc510464120)

[***TWEEPY* 15**](#_Toc510464121)

[***UNIDECODE* 15**](#_Toc510464122)

[**R LIBRARIES *LIBRARY(TM)* 15**](#_Toc510464123)

[***LIBRARY(IGRAPH)* 15**](#_Toc510464124)

[**Discussions of Module – 1: 17**](#_Toc510464125)

[**Discussion On Module – 2: 18**](#_Toc510464126)

[**Discussion On Module – 3: 18**](#_Toc510464127)

[**RESULTS: 19**](#_Toc510464128)

[**SUMMARY AND DISCUSSIONS: 22**](#_Toc510464129)

[**Future Scope: 23**](#_Toc510464130)

[**Conclusion: 23**](#_Toc510464131)

[**References: 23**](#_Toc510464132)

|  |  |  |
| --- | --- | --- |
|  |  |  |

# 

# INTRODUCTION:

People share their views and opinions on social media. These views can be collected and systematically arranged into a dataset. By analysis of this huge data, information about an event (a person’s opinion about it) can be identified. The views shared on the platform can be categorized as positive, negative or neutral.

For this project, we divided our work flow into three sub – tasks:

* **Extraction of data using twitter (Dataset development):** This requires the twitter API, for extraction of tweets from the twitter database. This will be the initial step for developing the dataset, which will be further used for analysis purpose
* **Analyzing the polarity and subjectivity of the dataset:** After obtaining the dataset, we’ll analyze the dataset for obtaining the polarity and subjectivity for our dataset. These are the matrices that are defined for tracking the sentiment involved in a tweet.
* **Visualization of the obtained Data:** In order to get a better insight of the obtained data one has to visualize the data, keeping our search as center point, what are the other search points or words that are also going popular with our search? We’ll also obtain new mathematical results using machine learning algorithms.

## ABSTRACT:

This project discusses the modern techniques used to identify and combat possible terrorist threats. It includes countering-terrorism by identifying possible affinity to terrorist activities. Various algorithms can be employed to detect abnormal activities. Opinion Mining is one such example that can be used to train systems to detect future pattern based on the model data set provided.

**Keywords:** Machine Learning, Opinion Mining, Analysis, Counter-Terrorism

## Problem Statement:

* Identifying affinity to terrorist activities.
* Pointing out the abnormalities in the day today activities.
* Applying appropriate classification techniques.
* Drawing conclusions on the basis of patterns in the classified data.

# LITERATURE REVIEW

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.NO. | PAPER TITLE | YEAR (PUBLICATION) | AUTHOR | SUMMARY |
| 1 | A NEURAL NETWORK FOR COUNTER-TERRORISM | AI-2011 Thirty-first SGAI International Conference on Artificial Intelligence, Cambridge, England. | M.B Dixon  J.Elliott  E. Guest  D.J. Muller  S.J Dixon | The paper covers the usage of neural networks to identify abnormal behavior. Experiments designed by psychologists and criminal enthusiasts were used for determining the  usefulness of implementing AI techniques to counter terrorism. The results showed the highest success rate of 68% to correctly indentify inconsistent behavior. Thus neural networks could be used to provide a good efficiency in predicting terrorism and would be helpful in implementing steps to counter the same. |
| 2. | FUTURE TERRORIST ATTACK PREDICTION USING MACHINE LEARNING ALGORITM | Working Paper-May 2017.  Affiliation: PESIT South Campus | SnehanshuSaha, HarshaAladi, Abu Kurien, AparnaBasu | This paper proposes the use of machine learning to detect patterns of behavior among terror outfits. It makes use of ensemble learning to detect the weapon systems used or the target locations by weighing out the strengths and weaknesses of every location and analyzing the maximum destruction-whether to life or property. It uses model test data to train the system and further incorporates concepts of statistics and regressions to train the system into calculating the future probabilities based in past experiences, thereby creating a smart learning system. |
| 3. | SOCIAL NETWORK ANALYSIS IN THE STUDY OF TERRORISM AND POLITICAL VIOLENCE | Southern Illinois University Carbonade(OpenSIUC)  Working papers from the Political Networks Paper Archive | AriePerliger, Ami Pedahzur | The article emphasizes how social network analysis is used to get information about characteristics of terror groups and organizations. It elucidates how the study of political violence is used to determine patterns, understand behaviors and predict outcomes of terrorist actions. This information provides insight on the division of power in the organization, the process of recruitment and the basic hierarchy all of which is potent in destroying the network. |
| 4. | SOCIAL NETWORK ANALYSIS AND COUNTER-TERRORISM: MEASURES OF CENTRALITY AS AN INVESTIGATIVE TOOL | Journal of  Behavioral Sciences of Terrorism and Political Aggression  Volume 5, 2013 - Issue 2: Applying Social Network Analysis to Terrorism | Sam Mullins | This article deals with the use of network analysis centrality measures as an investigative tool in determining the key players, spokesperson and hierarchy of various terror outfits. Pin pointing the most prestigious players can help track the people of importance and thereby shut down the network once and for all. |
| 5. | Detecting, Tracking and Counteracting Terrorist Networks via Hidden Markov Models | IEEE Aerospace Conference | Jeffrey Allanach  Satnam Singh  Peter Willett  HaiyingTu  Krishna Pattipati | The paper describes how hidden Markov Models can be used in modeling and detecting hidden terrorist networks. It includes gathering information from emails, usage mining, newspapers, subscribed links, financial institutions etc. First step is identifying terrorist threats followed by predicting their actions and finally predicting ways to counteract terrorism activities. |
| 6. | A NETWORK DESIGN APPROACH TO COUNTERING TERRORISM | The NPS Institutional Archive DSpace Repository | Torner  Linus P. | This thesis incorporates a comparative case study of network design approach against social network analysis to propose methods to stop future terrorism efforts. It aims to remove the balance of terrorist organizations and improve the efficiency of their algorithm to break into the terrorist community. |
| 7. | A Survey on Social Network Analysis for Counter-Terrorism | International Journal of Computer Applications  © 2015 by IJCA Journal  Volume 112 - Number 9  Year of Publication: 2015 | PankajChoudhary, Upasna Singh | The study aims at incorporating social network analysis tools to predict the hierarchical structure of major terrorist factions. The concepts of SNA elucidated the feasibility of online and offline communities and helped determine a pattern among the same. Predicting the nature of the organization provides insight into the psyche of its members and sheds light on their agendas, strongholds as well as weaknesses. All of which helps to infiltrates the network, predict their movements and hopefully stop the perpetrators from harming innocent civilians. |
| 8. | NETEST: ESTIMATING A TERRORIST NETWORK’S STRUCTURE | CASOS 2002 Conference | Matthew J. Dombroski,  Kathleen M. Carley | The paper proposes the implementation of Bayesian Probabalistic frameworks to produce accurate classifications of terrorist organizations. It involves combining informant information with standard Bayesian inferences. The paper plays out scenarios varying the area of the network, the membership structure, cost analysis to predict the regions under greater threat of impending terrorist activities. These “what-if” scenarios play a two-fold crucial role-1. To strengthen the scrutiny over a particular area, thereby increasing the chances of catching the perpetrators 2. To warn people and take necessary measures to protect them from impending danger. |
| 9. | PREDICTIVE ANALYSIS OF CONCEALED SOCIAL NETWORK ACTIVITIES BASED ON COMMUNICATION TECHNOLOGY CHOICES: EARLY-WARNING DETECTION OF ATTACK SIGNALS FROM TERRORIST ORGANIZATIONS | Computational and Mathematical Organization Theory  March 2010, Volume 16, Issue 1 | * Katya Drozdova * Michael Samoilov | The paper fuses technology with the classical social network analysis concepts to detect precursors of terrorist attacks. The kind of technology used can be used to provide insight into the organization’s activities, hierarchy and can be further used to identify and mitigate terrorist threats. |
| 10. | COLLABORATIVE MINING IN MULTIPLE SOCIAL NETWORKS DATA FOR CRIMINAL GROUP DISCOVERY | Twelfth International Conference on Computational Science and Engineering;2–6 March | Amin MilaniFard, Martin Ester | The paper proposed a framework for automated network data analysis and deduction approach from multiple social networks by converting to transaction dataset, applying association mining, and statistical methods. |

# *Social Media Analysis Techniques:*

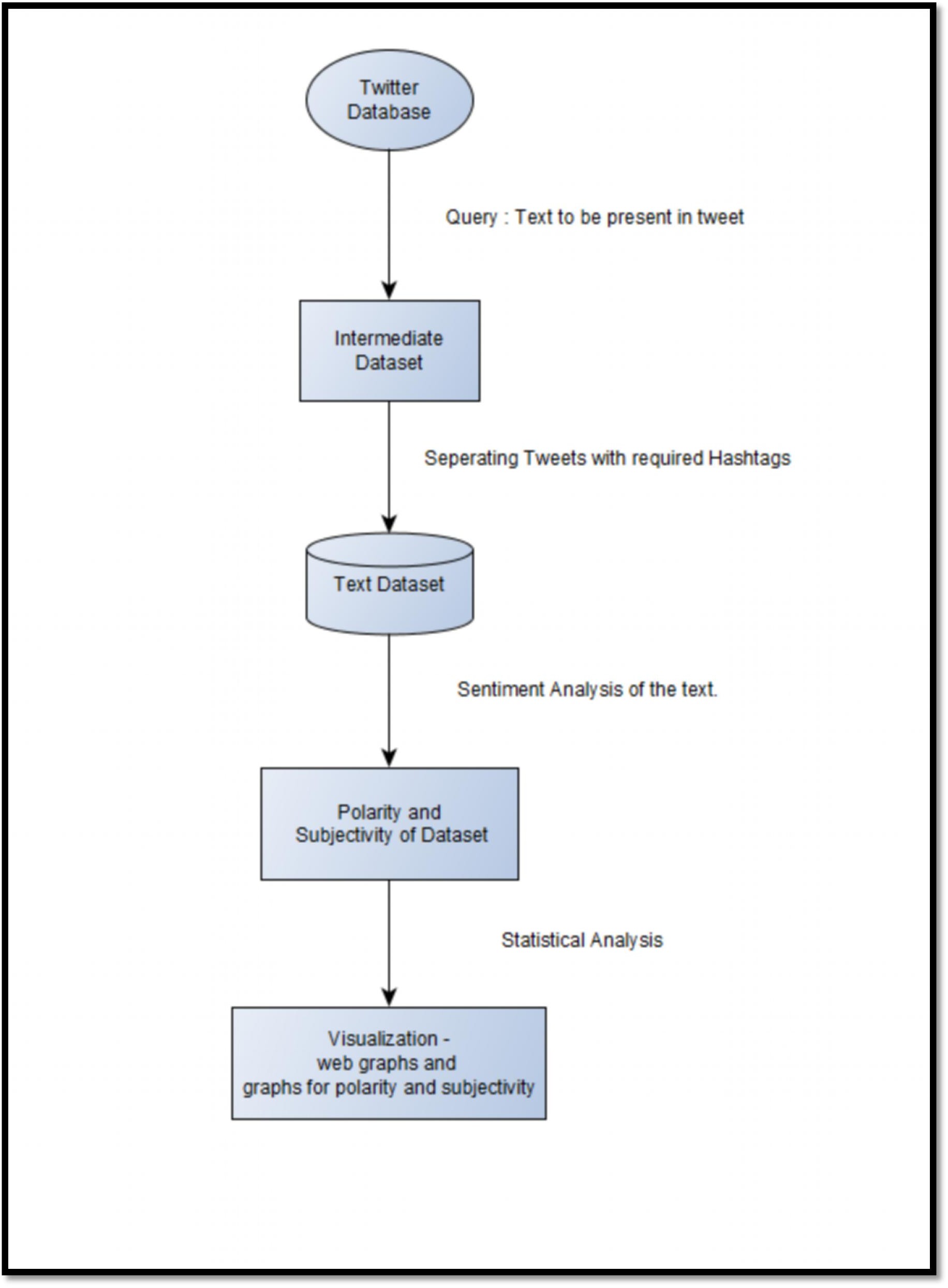
* Computational Science Techniques: This technique employs three broad fields- Computational Statistics ( resampling methods, Markov Chain , local regression and so on), Machine Learning (Regression Trees, Support Vector Machines, in supervised learning and Self Organizing Maps, K-Means in unsupervised learning) and Complexity Science (complex simulation models of difficult-to- predict systems that are derived from statistical physics, information theory and nonlinear dynamics).
* Sentiment Analysis: It refers to mining of emotions, attitudes and feelings. It is subjective rather than objective. It can be divided into subtasks- sentiment context, sentiment level, sentiment subjectivity, sentiment polarity, sentiment strength. Various statistical and machine learning algorithms which are used for sentiment analysis are- Naïve Bayes, Maximum entropy, Support Vector Machines, Logistics Regression Model and Latent Semantic analysis.

# *Social Media Analytics Tools:*

* + Scientific programming tools: Scientific analytics libraries and tools have been developed to provide support for sourcing, searching and analyzing text. R - used for statistical programming, MATLAB - used for numeric scientific programming, and Mathematica - used for symbolic scientific programming (computer algebra).
  + Business Toolkits: Business Toolkits are commercial suites of tools that allow users to source, search and analyze text for a range of commercial purposes. Examples include: SAS Sentiment Analysis Manager, RapidMiner, Lexalytics and IBM SPSS Statistics.
  + Text Analysis Tools: Text analysis tools are used for natural language processing and text analysis. Examples include: OpenAmplify, Jodange, LinePipe, Python NLTK.
  + Data Visualisation Tools: The data visualization tools allow users to gain insights from the big data. The users can perform analysis through interactive user interfaces available on the majority of devices. Examples include: SAS Visual Analytics and Tableau.

# Architecture / Methodology / Algorithm / Framework used:

**ARCHITECTURE:**



**METHODOLOGY:**

1. **Twitter Database**: Twitter Database is the live and dynamic database that is used for the mining of tweets based on our input query. We need to have a valid Authentication set of details in order to access with Twitter’s API.

Next an Intermediate Database is generated by the query we provide.

1. **Intermediate Database**: Intermediate Database is the data stream that is been extracted from twitter against our input query. It generates when we access the twitter API from our code.

After obtaining the database, database’s hash tags are matched with the required hash tags and the result is stored in a CSV format.

1. **Text Dataset**: Text Dataset is the customized version of intermediate database, which is obtained by matching the hash tag obtained from twitter data with our desired hash tag(s). This data base is stored in a CSV format.

After obtaining the required text dataset it is subjected for the sentiment analysis, by using supervised learning, we find Polarity and subjectivity of each tweet, and then find the average polarity for a query of the dataset.

1. **Polarity and Subjectivity of Dataset**: These are the values that are obtained by the sentiment analysis of each and every tweet that is present in our dataset, it is done by a package wrapper of Text Blob which uses supervised learning for obtaining the said values.

After generating polarity and subjectivity, the data is then processed for visualization, here we generate web graphs, other logical relations.

1. **Visualization of the data:** Visualization of data is performed using R language. Here we generate web graphs related to our query, and also infer some of the mathematical relations in – order to quantify the sentiments.

## ALGORITHM:

|  |
| --- |
| from textblob import TextBlob  import csv  import tweepy  import unidecode  import sys  reload(sys)  sys.setdefaultencoding('utf-8')  #(OAuth)  f = open('auth.k','r')  ak = f.readlines()  f.close()  auth1 = tweepy.auth.OAuthHandler(ak[0].replace("\n",""), ak[1].replace("\n",""))  auth1.set\_access\_token(ak[2].replace("\n",""), ak[3].replace("\n",""))  api = tweepy.API(auth1)  target\_num = int(raw\_input())  query = raw\_input()  hashtags=map(str,raw\_input().split())  print hashtags  csvFile = open('results\_social.csv','w')  csvWriter = csv.writer(csvFile)  csvWriter.writerow(["username","author id","created", "text", "retwc", "hashtag", "followers", "friends","polarity","subjectivity"])  counter = 0  for tweet in tweepy.Cursor(api.search, q = query, lang = "en", count = target\_num).items():  created = tweet.created\_at  text = tweet.text  text = unidecode.unidecode(text)  retwc = tweet.retweet\_count  try:  hashtag = tweet.entities[u'hashtags'][0][u'text']  print hashtag #hashtags used  except:  hashtag = "None"  username = tweet.author.name  authorid = tweet.author.id  followers = tweet.author.followers\_count  friends = tweet.author.friends\_count  text\_blob = TextBlob(text)  polarity = text\_blob.polarity  subjectivity = text\_blob.subjectivity  for i in range(0, len(hashtags)):  if(hashtags[i] == hashtag):  csvWriter.writerow([username, authorid, created, text, retwc, hashtag, followers,  friends, polarity, subjectivity])  counter = counter + 1  if (counter == target\_num):  break  csvFile.close() |

**CODE:**

1. **PYTHON CODE**

from textblob import TextBlob

import csv

import tweepy

import sys

reload(sys)

sys.setdefaultencoding('utf-8')

consumer\_key= 'KURJcvd3k9G19V1atOnF3m13N'

consumer\_secret= '5DxhcVZEaNgqr9mzzFZoJl2YXNOCJ1BnoMIzH70VLTFbt5I6KG'

access\_token='973866547977555968-UGhvNKQeh571n6h8j6MrSLE8pcFNkYe'

access\_token\_secret='t3Kfl53gULJj72LsfOfQytThpFGvpdcwKaxvBjz94U5vm'

auth = tweepy.OAuthHandler(consumer\_key, consumer\_secret)

auth.set\_access\_token(access\_token, access\_token\_secret)

api = tweepy.API(auth)

target\_num = int(raw\_input())

query = raw\_input()

csvFile = open('isis.csv','w')

csvWriter = csv.writer(csvFile)

csvWriter.writerow(["username","authorid","created", "text", "retwc", "followers", "friends","polarity","subjectivity"])

counter=0

for tweet in tweepy.Cursor(api.search,q=query,lang="en",count=target\_num).items():

created = tweet.created\_at

text = (tweet.text).encode('utf8')

retwc = tweet.retweet\_count

username = tweet.author.name

authorid = tweet.author.id

followers = tweet.author.followers\_count

friends = tweet.author.friends\_count

text\_blob = TextBlob(text)

polarity = text\_blob.polarity

subjectivity = text\_blob.subjectivity

counter=counter+1

csvWriter.writerow([username, authorid, created, text, retwc, followers, friends, polarity, subjectivity])

if(counter==target\_num):

break

csvFile.close()

1. **R CODE**

|  |
| --- |
| > library(tm)  > library(igraph)  > twitter\_data = read.csv("C:\\Users\\Asus\\Desktop\\ai\_project\\afghanistan.csv", header=TRUE,sep=",",encoding="UTF-8",stringsAsFactors=FALSE)  > twitter\_data$text = paste(substr(twitter\_data$text,2,nchar(twitter\_data$text)))  > text = twitter\_data$text  > text\_clean = gsub("(RT|via)((?:\\b\\W\*@\\w+)+)", "", text)  > text\_clean = gsub("@\\w+", "", text\_clean)  > text\_clean = gsub("[[:punct:]]", "", text\_clean)  > text\_clean = gsub("[[:digit:]]", "", text\_clean)  > text\_clean = gsub("http\\w+", "", text\_clean)  > text\_corpus <- Corpus(VectorSource(text\_clean))  > text\_corpus = tm\_map(text\_corpus, tolower)  > text\_corpus = tm\_map(text\_corpus, stripWhitespace)  > text\_corpus = tm\_map(text\_corpus, PlainTextDocument)  > text\_corpus = Corpus(VectorSource(text\_corpus))  > tdm = TermDocumentMatrix(text\_corpus)  > m = as.matrix(tdm)  > wf = rowSums(m)  > m1 = m[wf>quantile(wf,probs=0.9), ]  > m1 = m1[,colSums(m1)!=0]  > m1[m1 > 1] = 1  > termMatrix = m1 %\*% t(m1)  > library(igraph)  > g = graph.adjacency(termMatrix, weighted = T, mode = "undirected")  > g = simplify(g)  > V(g)$label <- V(g)$name  > V(g)$degree <- degree(g)  > set.seed(3535)  > layout1 = layout.fruchterman.reingold(g)  > plot(g, layout=layout1)  > V(g)$label.cex = 1.2 \* V(g)$degree / max(V(g)$degree) + 0.2  > V(g)$label.color = rgb(0.0, 0.0, 0.2, 0.8)  > V(g)$frame.color = NA  > egam = (log(E(g)$weight) + 0.3) / max(log(E(g)$weight) + 0.3)  > E(g)$color = rgb(0.5, 0.5, 0.0, egam)  > E(g)$width = egam  > plot(g, layout=layout1) |

# 

**R CODE FOR CLUSTERING**

> data=read.csv("C:\\Artificial Intelligence Project\\analysis\\graphicalRep.csv", header=TRUE,sep=",",encoding="UTF-8", stringsAsFactors = FALSE)

> head(data)

> library(ggplot2)

> ggplot(data,aes(polarity,subjectivity,color= Term)) + geom\_point()

> set.seed(20)

> dataCluster <- kmeans(data[,2:3],3,nstart = 20)

> dataCluster

> table(dataCluster$cluster,data$Term)

> dataCluster$cluster <- as.factor(dataCluster$cluster)

> ggplot(data,aes(polarity,subjectivity,color=dataCluster$cluster))+geom\_point()

# FRAMEWORK USED:

If we broadly classify our project, the workflow divides into the following two categories:

1. *Data extraction and sentiment analysis of the developed dataset.*
2. *Data visualization of the collected data.*

We are using state – of – the – art frameworks available for obtaining results for the same.

For the purpose of data extraction and sentiment analysis of the developed dataset we are using **Python v2.7.13**as our programming language**. We’re using PyCharm Community version 2017.1.2** as the development environment**.** Python is a general purpose programming language, which is known for its ultra – readable nature and a very preferred choice for developments in the fields of machine learning, AI community. It’s now the fastest growing programming language. PyCharm is the best suitable IDE for the required purpose.

Further details will be discussed in upcoming points.

For the purpose of Data Visualization of the developed dataset, we’re using,

**R version 3.4.1** as our programming language, and **RStudio version 1.0.153** as our development environment. R is the language that is developed mainly for analyzing the datasets. Obtaining various complex mathematical relations using just one command.

# LIBRARIES USED

**PYTHON LIBRARIES**

# *TextBlob*

It is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

# *CSV*

CSV (Comma Separated Values) format is the most common import and export format for spreadsheets and databases. The csv module implements classes to read and write tabular data in CSV format. Programmers can describe the CSV formats understood by other applications or define their own special-purpose CSV formats.

# *TWEEPY*

Python library for accessing the Twitter API.

# *UNIDECODE*

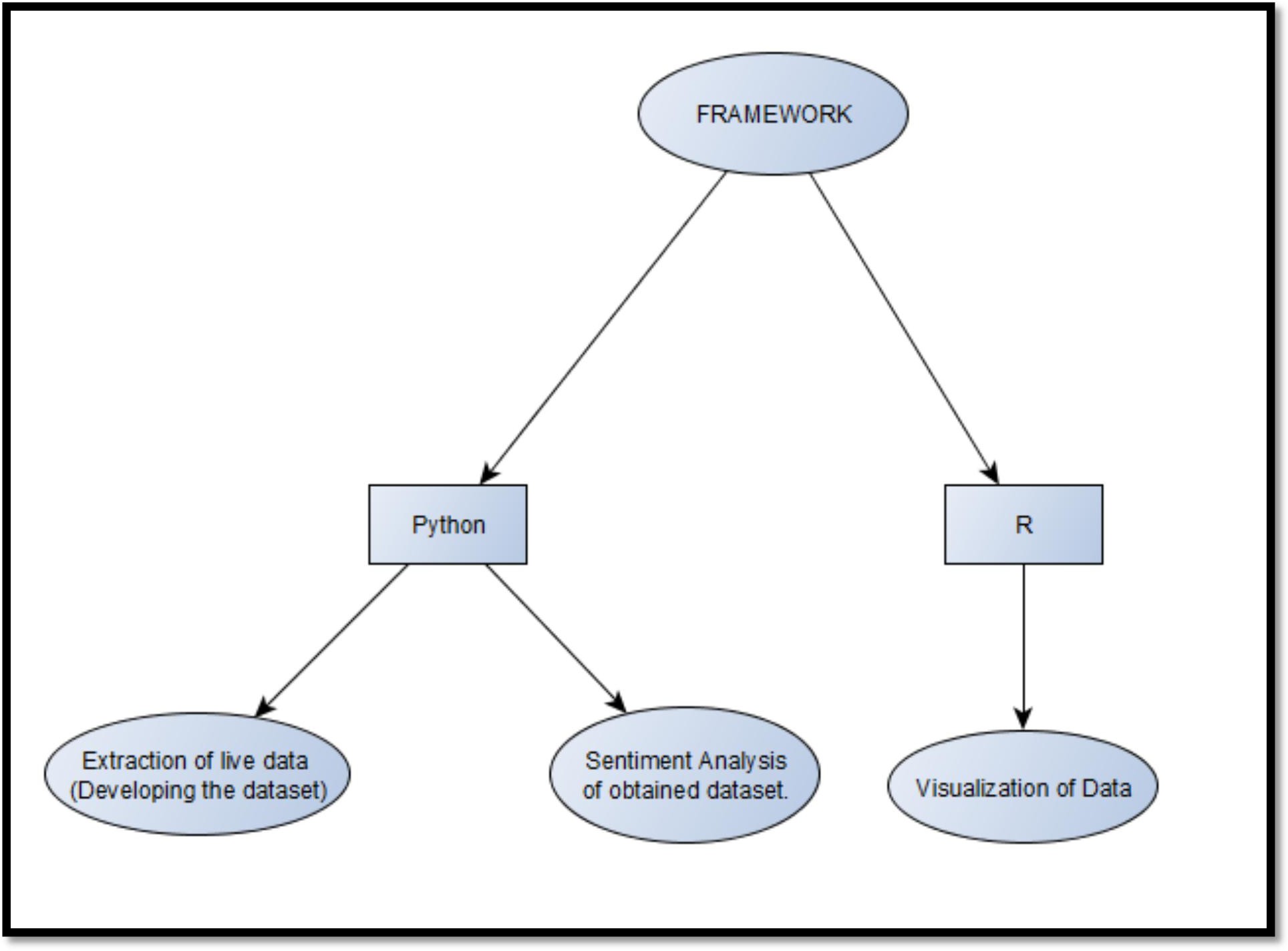
Unidecode is a middle road: function *unidecode()* takes Unicode data and tries to represent it in ASCII characters where the compromises taken when mapping between two character sets are chosen to be near what a human with a US keyboard would choose.

# R LIBRARIES *LIBRARY(TM)*

A framework for text mining applications within R. The main structure for managing documents in tm is a so-called Corpus, representing a collection of text documents. A corpus is an abstract concept, and there can exist several implementations in parallel.

# *LIBRARY(IGRAPH)*

Provides Routines for simple graphs and network analysis. It can handle large graphs very well and provides functions for generating random and regular graphs, graph visualization, centrality methods and much more.



**Fig, Workflow definition for the project, defining the frameworks.**

**Packages / dependencies that are used in Python and R:**

***Python:***

1. ***CSV package:*** for maintaining the dataset in csv format.
2. ***SYS package:*** for removing any language translation errors.
3. ***UNIDECODE package:*** for ease – of – decoding of data.
4. ***TWEEPY:*** The API client of twitter used for Extraction of data.
5. ***TEXTBLOB:*** The package that is used for calculating polarity and subjectivity.

**Packages / Libraries that are used in R:**

1. ***TM package :*** Package that is used to generate Corpus.
2. ***IGRAPH Package:*** Used for Network Analysis and visualization.
3. ***GGPLOT :*** Used for Basic graph plotting.

# Discussions of Module – 1:

***EXTRACTION OF DATA USING TWITTER:*** Data extraction from Twitter requires a valid twitter developer account, so that one can have access to the Twitter API. It provides us with four details:

* + Consumer Key
  + Consumer Secret
  + Access Token
  + Access Token Secret

Graphical user interface, application

Description automatically generated

**Fig, Authentication of Twitter’s Developer Account.**

All of these keys are highly confidential, and are not meant to be shared publically.

Tweepy is an easy – to – use Python Library for accessing the twitter data. We have to store all the details that are provided by twitter developer side in a text file that will be accessed by tweepy, as “**auth.k”.**

Tweepy uses the OAuth mechanism for authentication our program to access the twitter database for development purpose. It uses tweepy.API () function for that purpose.

After Authentication has been done, we have to enter the query on which we want to do sentiment analysis, perform data visualization on the same and also the no. of hash tags that we want to include in our dataset.

We develop our dataset in a CSV file format, by using “**csv”** package. We also have to limit the target no. of tweets that have to be stored under the dataset. As the database of tweets is live and dynamic.

Now after initializing a CSV file open and write request, we start extracting the tweets based on the cursor that moves around the live and dynamic twitter database of tweets. Now we start filling our CSV file after each iteration, initially we provide query and hash tags. Our hash tags can be a list of strings, which enables us to have a dataset of customized tweets. After reaching the required number of queries, the cursor stops crawling across the dataset. And hence we obtain our customized dataset.

# Discussion On Module – 2:

**OPINION ANALYSIS OF GENERATED DATA:**

After having developed our dataset, we computed **“Subjectivity”** and “**Polarity”.** Subjectivity and Polarity are the performance matrices that are used to evaluate the sentiments of the tweet.

We used **Text Blob** as our package for evaluation of tweets that are present in our dataset. Text Blob uses a trained Naïve Bayesian Classifier that is used to for evaluating polarity and subjectivity of the dataset. Text Blob uses NLTK packages internally in order to classify the given text.

**Polarity :** Polarity of a given entity (text/ sentence) is defined as the opinion of the text – that is whether the expression is carrying a positive opinion, a negative opinion or a neutral opinion, it ranges between [ –1.0 , 1.0].

After generating the polarity of the individual tweets for the whole dataset, we calculate the average polarity. Using this we infer what is the general reaction of mass public on a given situation.

**Subjectivity:** Subjectivity of a given entity (text/sentence) can be defined as deciding whether a given text expresses an opinion or is factual without expressing a positive / negative opinion.

For example: Floyd May weather won his 50th match and tweeted: 50 – 0. This is an objective tweet. However, “Floyd May weather won his final match, the streak goes on as 50 – 0. #congrats” is a subjective tweet.

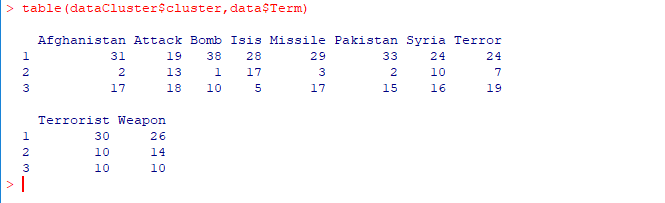
Polarity and Subjectivity are of very high importance in corporate world. They can be used for the companies to analyze the reviews for their products and shape their business models and investment plans accordingly.

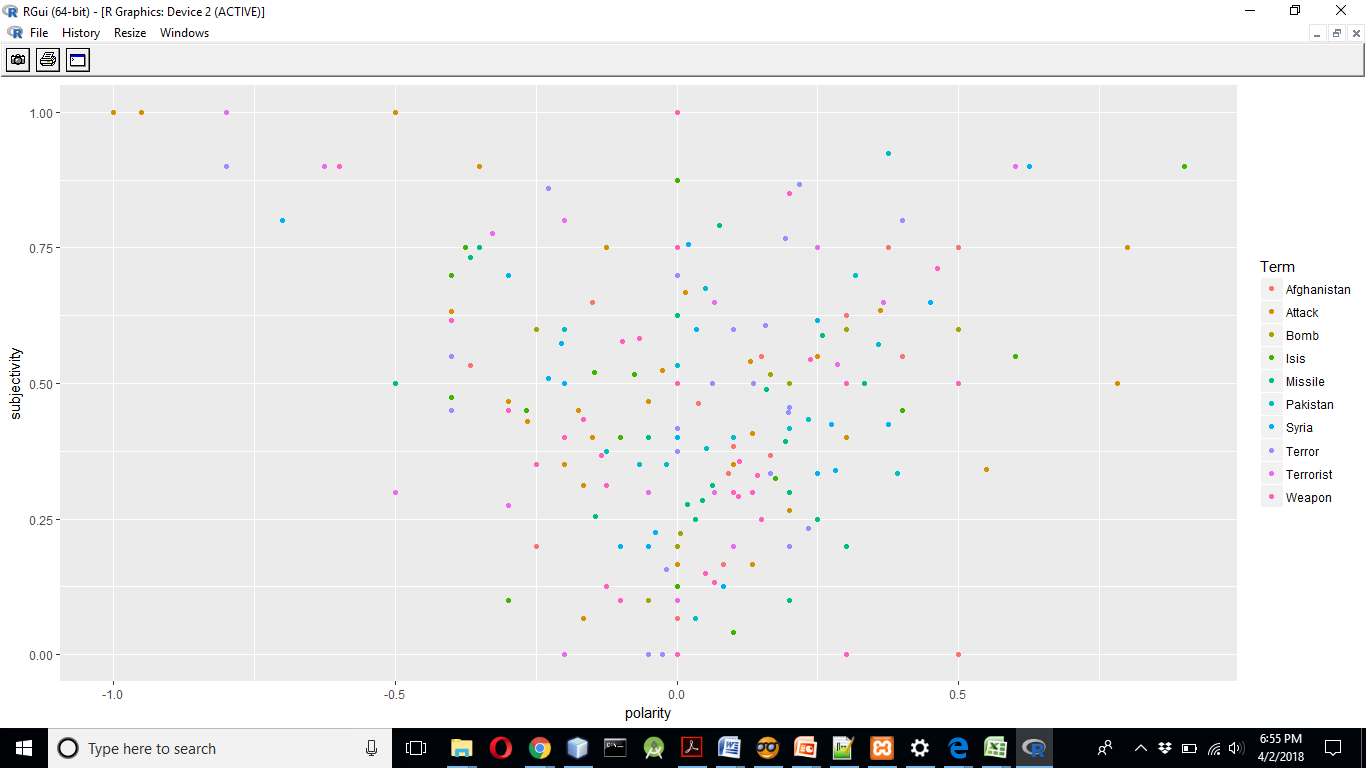
# Discussion On Module – 3:

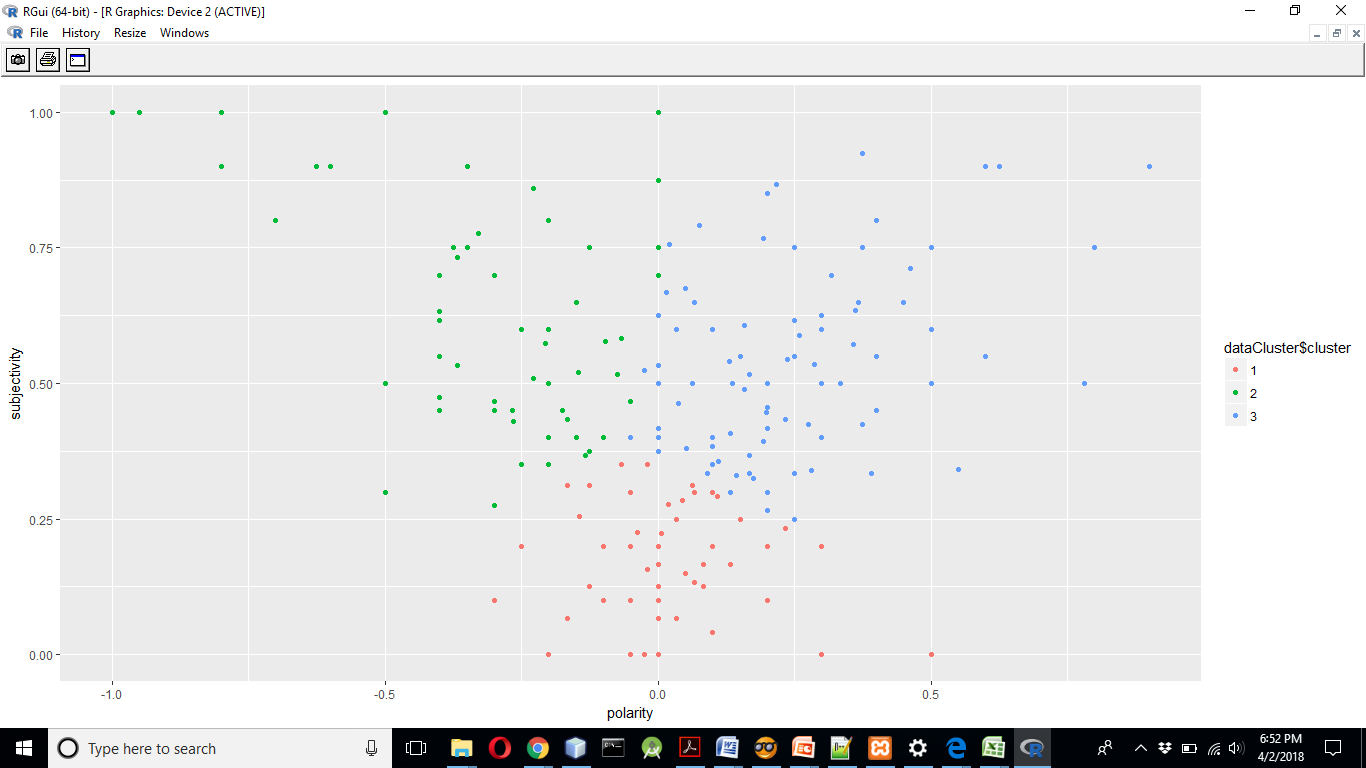
After sentiment analysis in module 2 from the dataset obtained from module 1, visualization of the datasets results in mathematical relationships among various terms and a web graph of the terms widely used with the input query. This provides a better understanding of people’s reaction to a situation (positive/negative) obtained by calculating the average polarity for certain query terms. On the basis of the obtained data further inference may also be possible.

# RESULTS:

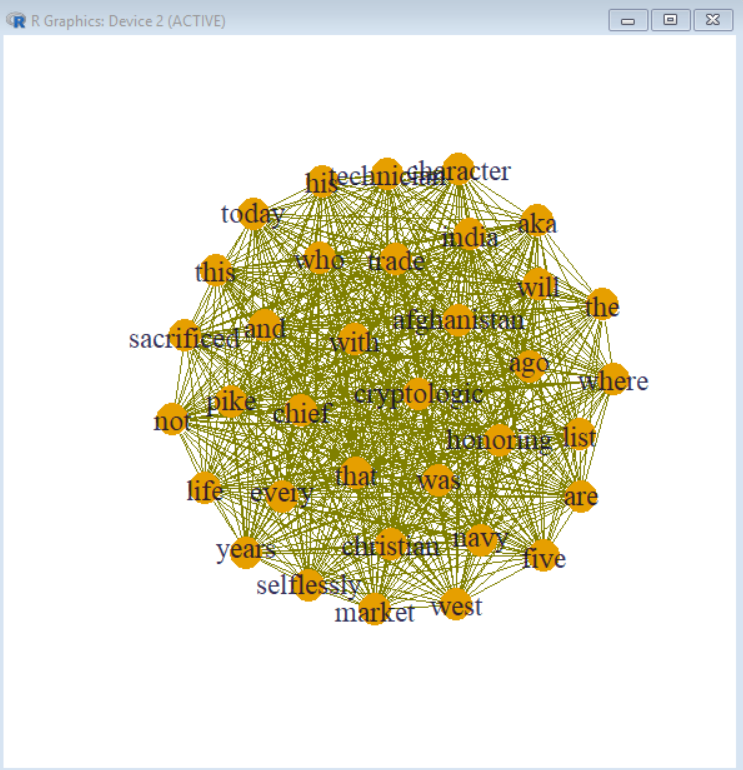




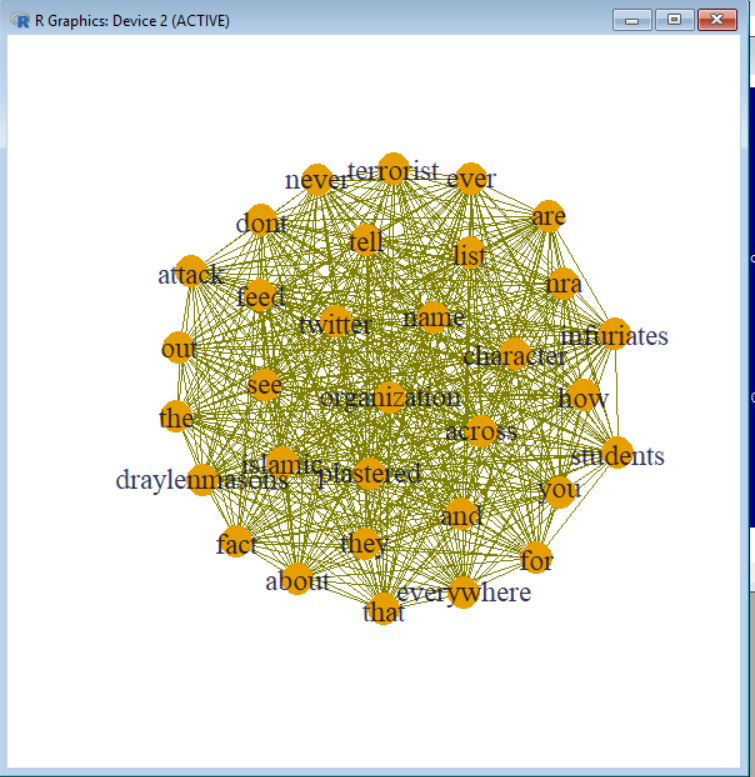




The word-graph generated to visualize how different words are connected to each other and how often do them appears in the same tweet.

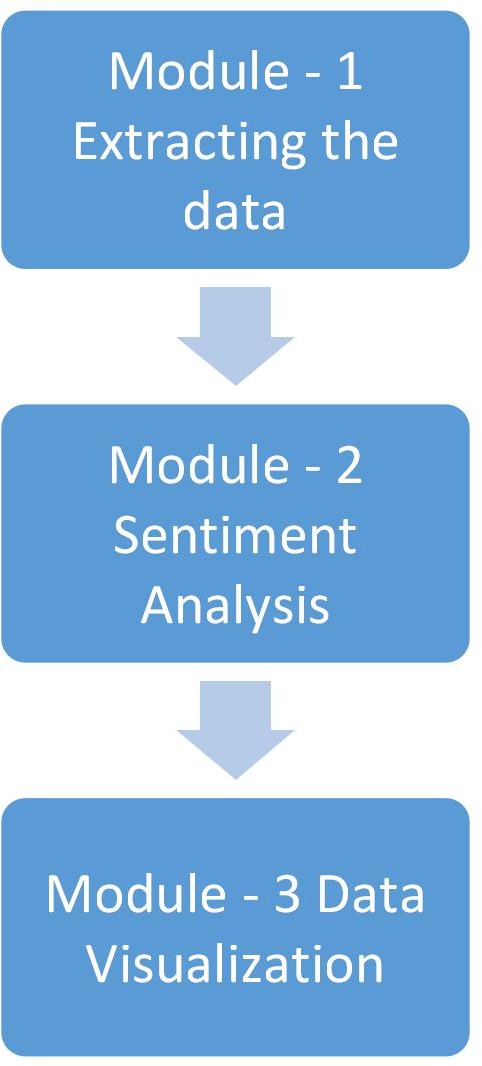


***AFGHANISTAN***



# SUMMARY AND DISCUSSIONS:

1. In module 1 we extracted the data from twitter database using OAuth Protocol (for verification of the developer), then processed the intermediate database against our required set of hash tags, and hence were able to manipulate the data according to our terms. The extraction was done by tweepy package, and the maintenance of the CSV file was done by CSV package to ensure that no language encoding problem happened while updating the dataset. Unicode and sys packages were also used.
2. In module 2 the extracted data was used to calculate the polarity and subjectivity of each and every tweet of the database. This was done using the text blob package of python, which uses supervised learning and has a trained naïve Bayesian classifier which calculates the values of the polarity and subjectivity.
3. In module 3 mathematical relationships among various terms was calculated by data visualization of our dataset and a web graph for the same was obtained. This provided a better understanding of people’s reaction (positive/negative) to different situations.



## FUTURE SCOPE:

Till now we are done with Data Gathering and Opinion mining. Now we shall apply machine learning algorithms and graphical representation algorithms to predict persons affinities to the terrorism behavior.

## CONCLUSION:

The main aim of this paper was to emphasize on the adaptation of new methods in countering terrorism. It enunciates the benefits of using modern technology in pattern analysis, social networks and system training by providing information to give faster and accurate results.

## REFERENCES:

[1]Iskandar, B. S. (2017). Terrorism Detection Based on Sentiment Analysis Using Machine Learning. Journal of Engineering and Applied Sciences, 12(3), 691-698.

[2] A. Malathi and Dr. S. SanthoshBaboo, Evolving Data Mining Algorithms on the Prevailing Crime Trend An Intelligent CrimePredictionModel, In International Journal of Scientific & Engineering Research,June 2011, Vol. 2, Issue.

[3] Ana Swanson Wonkblog, The eerie math that could predict terroristattacks, March 1, 2016.

[4] V.S. Subrahmanian ,Terrorist Social Network Analysis: Past, Present,and Future UMIACS University of Maryland.

[5] Karstoft, K. I., Galatzer-Levy, I. R., Statnikov, A., Li, Z., &Shalev, A. Y. (2015). Bridging a translational gap: using machine learning to improve the prediction of PTSD.*BMC psychiatry*, *15*(1), 30.