

Emotive Sentiment Analysis Using Spark

**EMOTION DETECTOR ON THE BASIS OF REAL TIME DATA
FROM SOCIAL MEDIA CONTENT**



Developed By

Danyal Javeed

1867-FBAS/BSSE/F12

Abdul Malik

1856-FBAS/BSSE/F12

BS Software Engineering

Supervised By

Dr. Jamal Abdul Nasir

Department of Computer Science & Software Engineering
Faculty of Basic and Applied Sciences

International Islamic University, Islamabad

(2016)

**DEPARTMENT OF COMPUTER SCIENCE & SOFTWARE ENGINEERING
INTERNATIONAL ISLAMIC UNIVERSITY, ISLAMABAD**

FINAL APPROVAL

Dated: _____

It is certified that we have read the project report submitted by **Mr. Danyal Javeed (1867-FBAS/BSSE /F12)** and **Mr. Abdul Malik (1856-FBAS/BSSE/F12)** and it is our conclusion that this project is of sufficient standards to warrant its acceptance by the International Islamic University, Islamabad for the BS Degree in Software Engineering.

COMMITTEE

EXTERNAL EXAMINER

Mr. Muhammad Nadeem

Assistant Professor,
Department of Computer Science & Software Engineering
IIUI, Islamabad

INTERNAL EXAMINER

Mr. Asim Munir

Assistant Professor,
Department of Computer Science & Software Engineering
IIUI, Islamabad

SUPERVISOR

Dr. Jamal Abdul Nasir

Assistant Professor,
Department of Computer Science & Software Engineering
IIUI, Islamabad

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dedication

We would like to dedicate over research work to the HOLIEST man Ever Born on Earth,

PROPHET MUHAMMAD (Peace Be Upon Him)

And

We also dedicate over work too over

PARENTS & FAMILY

Whose sincere love and prayers were a source of
Strength for us and made us to do this research work
Successfully.

Declaration

We hereby declare that this Software, neither as a whole, nor as a part thereof has been copied out from any source. It is further declared that we have developed this software entirely on the basis of our personal efforts made under the sincere guidance of our teachers and supervisor. No portion of the work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Mr. Danyal Javeed

1867-FBAS/BSSE/F12

Mr. Abdul Malik

1856-FBAS/BSSE/F12

Acknowledgement

All praise to Almighty Allah, who gave us the understanding, courage and patience too complete This project.

Thanks to our parents and all well-wishers, who helped us during our most difficult times and it Is due to their untiring efforts that we are in this position today.

We express our gratitude to our kind teacher **Dr. Jamal Abdul Nasir** for providing us the opportunity to learn and enhance our knowledge. He had been ready to help and guide us throughout the project.

It is surely due to our parent's moral and financial support during over entire career that enables us to complete our work. We are also thankful to our loving brothers, friends who mean most to us.

Project in Brief

| | |
|------------------------|--|
| Project Title | EMOTIVE SENTIMENT ANALYSIS USING SPARK (ES AUS) |
| Objective | The primary intent of the project is to analyze the mood globally and compute the contents of real time twitter streaming using a spark engine. Monitor Stream Like Fear, Happy, Sad, Joy |
| Undertaken by | Danyal Javeed Abdul Malik |
| Supervised By | Dr. Jamal Abdul Nasir |
| Date Started: | November 2015 |
| Date Completed: | August 2016 |
| Tools Used: | Spark Streaming, Spark Engine, Spark M-Lib, Robo-Mongo DB, Elastic search, Elastic search Plugin-Head, Kibana, Net Beans IDE |
| System Used: | Sony Vaio Core i5 Windows 10 Toshiba Core i5 Windows 10 |

Abstract

Social networking has become a salient part of our information media. It affects the beliefs, values, attitude of people, as well as their behaviors. Therefore, converting information media content into a meaningful information, key concepts and themes is vital for generating knowledge and formulating strategies. Large scale analysis on information media content allows for real time discovery of macro-scale token pattern in public opinion and sentiment. In this research, we analyze a collection of 26 thousand tweets generated by more than 11 thousand users around the globe in a real time.

The analysis of news content is traditionally based on the coding of data by human coders. Now measuring the current public mood is a challenging task on a huge collection of data in real time. Public media such as Twitter or Facebook can easily become a valuable source of information about the public due to the fact that people use them to express their feelings on public media. It is feasible to capture the public mood by monitoring the stream of Twitter data. Our main focus is to track four moods which are **Fear, Joy, Happy** and **Sadness**. For each mood, we track a long list of associated words and we count the frequencies that these words appear in tweet. We describe how the analysis of Twitter content can reveal mood changes in entire populations, most crucial this survey aims to demonstrate some of the steps that can be automated, allowing researchers to access a macroscopic token pattern that would be otherwise out of reach.

Contents

| | |
|---|--------------|
| Chapter 1..... | |
| Introduction: | 0 |
| 1.1 Importance of Emotion Data Analytics. | 2 |
| 1.2 Aim and Objective:..... | 4 |
| 1.3 Description of The Data: | 4 |
| 1.4 Motivation: | 5 |
| 1.5 Document Structure and Contribution: | 6 |
| Chapter 2 | 6 |
| 2. Related Work: | 7 |
| 2.1 Results, Outcome: | 7 |
| 2.1.1 Twitter Mood | 7 |
| 2.1.2 Flu Detector | 8 |
| 2.1.3 Earthquake Analysis | 9 |
| 2.2 Other Events Prediction..... | 10 |
| 2.2.1 Election | 10 |
| 2.2.2 Box Office Revenues for Movies: | 10 |
| 2.3 Cricket Prediction Analysis | 11 |
| Chapter 3 | 12 |
| 3. Introduction: | 13 |
| 3.1 Historical Data: | 13 |
| 3.1.1 Data Collection: | 14 |
| 3.1.2 Data Filtration: | 14 |
| 3.1.3 Features Construction: | 15 |
| 3.1.4 Training and Classification: | 21 |
| 3.1.5 Testing Data: | 21 |
| 3.1.6 Results, Outcome: | 21 |
| 3.2 Social Media Data:..... | 21 |
| 3.2.1 Characteristics of Tweets:..... | 22 |
| Steps involved developing the decision making system via twitter | 23 |
| 3.2.2 Data Collection: | 24 |

| | |
|---|----|
| 3.2.3 Data Filtering: | 24 |
| 3.2.4 Normalization and Feature Reduction: | 25 |
| 3.2.5 Training and Classification: | 25 |
| 3.2.6 Testing Data | 25 |
| 3.2.7 Prediction Outcome | 26 |
| Chapter 4 | 27 |
| 4. Introduction..... | 27 |
| 4.1 Implementation of Historical Data..... | 27 |
| 4.1.1 Data Collection..... | 27 |
| 4.1.2 Data Filtration | 27 |
| 4.1.3 Feature Construction..... | 28 |
| 4.1.4 Training and Classification | 28 |
| 4.1.5 Testing Data | 29 |
| 4.1.6 Prediction outcome | 29 |
| 4.2 Implementation of Social Media Data | 30 |
| 4.2.1 Data Collection | 30 |
| 4.2.2 Data Filtering..... | 32 |
| 4.2.3 Normalization and Feature Reduction | 33 |
| 4.2.4 Machine Learning Methods | 37 |
| 4.3 Prediction Through Streaming Tweets | 40 |
| 4.3.1 Spark Streaming | 40 |
| 4.3.2 Apache Spark..... | 44 |
| 4.3.3 Apache Spark Core Libraries..... | 45 |
| 4.3.4 Apache Spark Processing On Data UI..... | 45 |
| 4.3.5 Apache Spark Processing On Spark Job..... | 46 |
| 4.3.6 Apache Spark Processing on Spark Executors..... | 47 |
| 4.3.7 Apache Spark Streaming with Twitter..... | 47 |
| 4.3.8 Apache Spark Processing Completed Jobs..... | 48 |
| 4.3.9 Apache Spark Environment Setting..... | 49 |

| | |
|---|-----------|
| Chapter 5 | 50 |
| 5. Conclusion | 50 |
| 5.1Future Work | 51 |
| 6. Chapter 6 | 54 |
| 6. Introduction | 54 |
| 6.1 Methodology | 54 |
| 6.2 Graphs | 55 |
| 6.2.1 Real Time Streaming in Seconds | 55 |
| 6.2.2 Emotion Results Using Different Graphs | 56 |
| 6.2.3 Emotion Results Using Pie Charts and Line Charts | 57 |
| 6.2.4 Tweets Graphs by Language | 58 |
| 6.2.5 Total Tweet Language Result Statuses. | 58 |
| 6.2.6 Tweet Graphs by Continent | 60 |
| 6.2.7 Kibana Dashboard..... | 62 |
| 6.2.8 Elastic Search Plugin-Head..... | 62 |
| Chapter 7 | 63 |
| 7. Introduction | 63 |
| 7.1Methodology | 63 |
| 7.2Model Performance Testing (Visualization) | 64 |
| 7.3Naïve Bayes Algorithm (Visualization) | 64 |
| References | 66 |

Chapter 1

Introduction

1. Introduction:

Information media allows for the easy gathering of the huge amount of data generated by the public through communicating each other. Twitter is a popular and valuable source of researchers, which gives access to real time data that is suitable for the analysis of public sentiment on a large scale. This type of analysis is of interest because it avoids self-reporting and opinion polling and therefore opens the possibility to access much larger population. On that this task can only be achieved with text mining and pattern matching technologies. The data from public media is extracted for analysis from information media. I.e., the data based on public opinion. We are analyzing Public opinion from Twitter. Twitter is a novel micro-blogging platform launched in 2006 with more than 25 million unique monthly visitors. On Twitter, any user can publish a short message referred to as tweet with a maximum length of 140 characters, which is visible on the public display. The public timeline conveying the tweets of all users worldwide is an extensive real-time information stream of more than one million messages per-hour.

We make use of standard tools for emotion detection, and we apply the dataset of 100,000 tweets collected from worldwide at a real time. Our main goal is to check, if the effects of social events can be seen in tweets and to speculate if some of them could even be predicted. In the first part of our analysis, which provides a sanity check, where we see that word counting approach can provide a reasonable approach to sentiment or emotion analysis. While this approach is very popular in the analysis of the text and our assumption is that this first method is vital part of the analysis. This method can be done by making the stream of words that relate to the sentiment of joy, fear, happy and sad. We observe some important events such as ‘Eid, Month of Ramadan, Independence

Day evokes the stream of happiness and joy in the Muslim population from all over the world especially in Pakistan.

The second and important part of our analysis is that where we see the visible changing in results of emotions. For example, if the government implements new laws for the people of that country. Now at that time, it's very valuable talk and share his opinions among the people of that country. Our results show the change points in reality and that its effects can still be observed. In other words, public mood still has not recovered from that announcement. We collect tweets from all over the world and show results by slicing sub-continent wise, language wise, country wise, and percentage of higher and lower results of emotions in real time streaming. We also compare results between these different countries and show these results in one graph.

1.1 Importance of Emotion Data Analytics.

In today's world, Twitter makes it easy to engage users and communicate directly with them, and in turn, users can provide word-of-mouth marketing for companies by discussing the products. Apache Spark is an open-source framework that allows to process big data in a distributed environment across clusters of computers using simple programming models and run on top of Hadoop and store data on a Hadoop distributed file system (HDFS). It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Spark performs data parallel computation. With massive advances in cheaper and reliable storage technologies, data about every emotion is being stored in such a way that the entire chain of events could be retrived. With such advancements in technology and huge stake involved in evoking events, the emotions data recorded is analyzed and converted into actionable knowledge by scientists to gain advantage over their policies for public as an official level.

Twitter is a powerful source of social media that have generated millions of tweets in a second. Twitter provides a credential key that is used to access twitter web server that provides a **one** percent of whole server data. Data that are available on the server is in a raw form, but this data like text known as tweet can be used to extract useful information through data filtration and using different **NLP** (Natural language Processing) algorithms like POS tagger naïve bays linear regression etc.

Twitter has a valid information gathering source from all over the world because every country in the world allowed to every citizen that can create accounts on twitter and share information with one another. One tweet can provide useful information through analysis and produce a proper outcome that covers your need. Similarly, every company that knows more about Big data are currently working on Twitter data to extract useful information for different purposes.

Global analysis that can be already performed on twitter data in 2012-2015 are earthquake detection, flu detection, movie ratings, App rating, cricket rating, election in the USA and a few other states. All these above analyses that have been declared useful outcome are encouraging us to further work on human moods at a real time.

So we would like to discuss the importance of over project in real life at a real time. We are using four basic moods of humans like happy, sad, fear, joy using twitter data to detect around globe people's feelings. Accessing twitter data through the API and query data that's having proper location and language for emotive analysis on the tweet.

The analysis of human mood in a certain country or language specific provides human feelings that are extracted from tweets. Every tweet doesn't provide useful information because some tweets or irrelevant and some are neutral, but most tweets provide specific information. Tweets

Using hashtags declared the real time famous trending on twitter. Every country people feeling can be detected through our analysis and check the situation by monitoring the stream of twitter. Also check the trend (Events) that affects the feelings of people and common in our results outcome.

1.2 Aim and Objective:

The primary aim of this project is to establish a consistent statistical approach to calculate the outcome of the human moods. The main intent of the project is to analyze the mood globally and compute the contents of real time twitter streaming using a spark engine.

Like Fear, Happy, Sad, Joy.

Depends on a number of factors related to word count as well as the lexicon approach and synset of words. While some of these factors have been investigated in the literature, others have yet to be explored.

The following objectives were set to achieve the aims:

- To develop a dataset containing vital attributes that define the emotional outcome.
- To calculate the result of the human emotions in tweets at a real time.
- To check the outcome of results why people are happy, fear, joy and so on.

1.3 Description of The Data:

The concept is simple, where Twitter provides a platform for users on which they can publish brief tweets in a form of textual communication of up to 140 characters which are publicly accessible and easy to extract this information from public media. The communication or tweets tend to be in the moment expression of the user's current experiences.

Using the Twitter streaming API, we collected approximately 26,000 tweets from past and real Time. Then we periodically extract the 100 most recent tweets every 3-5 minutes, Geo-located to

Within a range of an urban Centre without specifying any keywords or hash tags. The text of each tweet stemmed by applying the **Stanford NLP POS tagger API** and Naive Bays Algorithm. The emotions or expressions are calculated by counting the frequency of emotion related words in each text published on a given day. The emotions related words were organized into four lists: Fear, Happy, Sadness, Joy. These word lists were based on extracts taken the priori base on WordNet API [9]. The list was further processed via stemming and filtering in a way that only single words are kept in the end. After these processing the word list contained 134 fear words, 84 sad words, 324 joy words, 112 happy words.

We compute the score of each word by emotion and expressions by scoring each word in the respective list, as the fraction of tweets containing on that day. We then aggregate this quantity over all words link to that particular emotion to obtain a score for each emotion on that day. This method is based on the assumption that average frequency of a word indicates its importance.

However, because the tweets are usually shorter and more ambiguous, sometimes it is not enough to consider only the current tweet for sentiment classification. We propose to improve target-dependent Twitter sentiment classification by 1) incorporating target-dependent features; and 2) taking related tweets into consideration. According to the experimental results, our approach greatly improves the performance of target-dependent sentiment classification.

1.4 Motivation:

Social media such as Twitter or Facebook, can easily become a valuable source of information about the public due to the fact that people use them to express their feelings and emotions in public. To capture the public mood by monitoring the stream of any social media. Tracking four moods which are “Fear”, “Joy”, “Anger”, “Sadness”.

Each mood we are tracking a long list of associated words and we count the frequencies that these words appear in the comments or tweets. This study can be used to benefit the people of the world and also helpful government officials of a country who want to know the real expression of the public in current affairs. This approach can be very helpful to take any decision by checking the expression of public in a new policy of a government. This approach is to measure the state of a population holds great promise for social scientists, data scientists perhaps also anthropologists. Here we have demonstrated how a simple experiment on a large amount of twitter data can reveal an important shift in public sentiment corresponding with the effects of the recession and government policy in this globe.

1.5 Document Structure and Contribution:

This document is organized as follows: This chapter contains a preliminary introduction of **Twitter Streaming API** and fully describes its purpose. The chapter also discusses brief history and some fundamental standard rules for text mining and pattern matching. In the next chapter, chapter 2, presents an overview of related work found in literature.

Chapter 2

Literature Review

2. Related Work:

Emotion detection analytics have direct applications in understanding and calculate human expression year by year on a large number of important events like ‘Eid, Ramadan, Independence Day and Christmas, etc. The complete set of data recorded for every event is proprietary to the owner of a profile or public media (Twitter). But a handful of high level Twitter data is accessible to the developers through websites like <https://dev.twitter.com/> (Twitter Developers), etc.

Human emotion extraction is a very valuable source to analyze human behavior, but a lot of work is also being done in different fields like Predicting the outcome of the Election, predicting box-office revenues for movies and many others.

In Section 2.1 below, we discuss some relevant work in the direction of modeling, simulation results and outcome of emotions.

In Section 2.2, we discuss some relevant work in the direction of modeling, simulation results and outcome of other different fields.

2.1 Results, Outcome:

The problem with the result, the outcome has been investigated in the context of Twitter Mood, Flu Detector and Earthquake.

2.1.1 Twitter Mood

Large scale analysis of social media content allows for real time discovery of macro-scale patterns in public opinion and sentiment. On that paper, researchers analyze a collection of 484 million tweets generated by more than 9.8 million users from the United Kingdom over the past 31 months, a period marked by economic downturn and some social tensions. Our findings, besides

corroborating our choice of method for the detection of public mood, also present intriguing patterns that can be explained in terms of events and social changes. On the one hand, the time series, we obtain show that periodic events such as Christmas and Halloween evoke similar mood patterns every year. On the other hand, we see that a significant increase in negative mood indicators coincides with the announcement of the cuts to public spending by the government, and that this effect is still lasting. We also detect events such as the riots of summer 2011, as well as a possible calming effect coinciding with the run up to the royal wedding [1]. The key idea is that by using a technique called Attribute Focusing, an overall distribution of an attribute is compared with the distribution of this attribute for a subset of data (e.g., Happy, Sad, Joy, Fear, entire season, etc.). If it has a characteristically different distribution for the focus attribute, it is marked as interesting. Such interesting patterns are discovered and provided to the domain expert to investigate further and gain insights.

2.1.2 Flu Detector

Tracking the spread of an epidemic disease like seasonal or pandemic influenza is an important task that can reduce its impact and help authorities plan their response. In particular, early detection and geolocation of an outbreak are important aspects of this monitoring activity. Various methods are routinely employed for this monitoring, such as counting the consultation rates of general practitioners. We report on a monitoring tool to measure the prevalence of disease in a population by analyzing the contents of social networking tools, such as Twitter. Our method is based on the analysis of hundreds of thousands of tweets per day, searching for symptom-related statements, and turning statistical information into a **flu-score**. We have tested it in the United Kingdom for 24 weeks during the **H1N1 flu pandemic**. We compare our flu-score with data from the Health

Protection Agency, obtaining an average a statistically significant linear correlation which is greater than 95%. This method uses completely independent data to that commonly used for these purposes, and can be used at close time intervals, hence providing inexpensive and timely information about the state of an epidemic [2]. This is considered to be an important decision in the context of flu detector.

2.1.3 Earthquake Analysis

Earthquake of large magnitude can often be classified as great natural catastrophes. This is usually the case when thousands of people are killed, hundreds of thousands are homeless. In the world the most important ones in terms of loss of lives were the 1976 Tangshan earthquake (China) with 290000 fatalities and 1970 Chimbote earthquake (Peru) 67000 fatalities. In terms of economic loss, the most important ones were 1995 Kobe earthquake (Japan) US \$100 billion and the 1994 Northridge earthquake (USA) with US \$44 billion. The 2001 Gujarat earthquake is a recent example of catastrophe. It was the first major earthquake to hit an urban area of India in the last 50 years. It Killed 13800 people, injured 167000 and a large number of reinforced concrete multistoried frame buildings were heavily damaged and many of them were collapsed completely in the towns of Kachchh district. Destruction total estimated to be about US\$ 5billion. The opinion that designing new buildings to be earthquake resistant will cause substantial additional costs is still among the constructional professionals. In a Swiss survey estimate between 3 and 17% of the total building costs were given. This opinion is unfounded. In a country of moderate seismicity adequate seismic resistance of new buildings may be achieved at no or no significant additional cost. [3]

2.2 Other Events Prediction:

The problem of outcome prediction has been investigated in the context of Election and Box-office revenues for movies.

2.2.1 Election

Malhar Anjaria, Ram Mahana Reddy Guddeti et al. Has predicted results for the US Presidential Elections-201 2 and Karnataka State Elections2013. For prediction they worked on user tweets and their opinions that are regarding the election. To collect data of the election, they use various hashtags like #USElections201 2, #USElections, #Elections2012 for US Presidential Election 2012 for the time period of August, 2012 to October, 2012. They have used Naive Bayes, SVM, MaxEnt, ANN. They used the unigram, bigram and a Unigram + Bigram (hybrid) feature extraction method for study purpose. Hybrid features are taken for absolute positive words like "wonderful", "awesome", "always" etc. and negative words such as "never", "not", "hardly" etc.

Nugroho Dwi Prasetyo and Claudia Hauff et al. Also work on election prediction and he predicts the Indonesian presidential elections in 2014. Using the specific case of the 2014 Indonesian presidential election, they aim to provide a detailed & in-depth analysis, comparing the Twitter-based prediction accuracy to 20 polls conducted by the most well-known polling institutes in Indonesia during the election campaign.

2.2.2 Box Office Revenues for Movies:

Sitaram Asur and Bernardo A. Huberman et al. has predicted box-office revenues for movies using the chatter from Twitter. They studied how sentiments are created, how positive and negative opinions propagate and how they influence people. Their focus on the topic of movies, is of

considerable interest among the social media user, and the real-world outcomes can be easily observed from box-office revenue for movies.

2.3 Cricket Prediction Analysis

One of the earliest and pioneering work in cricket was by Duckworth and Lewis where they introduce the Duckworth-Lewis or D-L method, which allows for fair adjustment of scores in proportion to the time lost due to match interruptions (often due to adverse weather conditions such as rain, poor visibility etc.). If the interruptions occur during the second innings, the team batting second will have less batting time and will face fewer balls. This affects the equilibrium and puts the second team at a disadvantage. To mitigate this, the target for victory has to be adjusted in proportion to the time lost. The D-L method is based upon a mathematical formulation that abstracts every ball and wicket of an ODI match into a single scalar called resource. Using this formulation, the number of overs lost is evaluated as runs and used to reset targets for the second team. This proposal has been adopted by the International Cricket Council (ICC) as a means to reset targets in matches where time is lost due to match interruptions. The method proposed in and subsequently adapted from for capturing the resources of a team during the progression of a match has found independent use in subsequent work in cricket modeling and mining.

Michael Bailey and Stephen R. Clarke uses historical match data and predict the total score of an innings using linear regression. As data of a match in progress streams in, the prediction model is updated. Using this, they analyze the betting market's sensitivity to the ups and downs of the game. Their model predicts the total score as instantaneous match data is streamed in.

Stylianios Kampakis and William Thomas et al. They worked on historical data and develop machine learning models in order to predict outcomes of the English twenty over county cricket cup over the years 2009-2014. They used a multi-step approach to analyze the data that produced more than 500 features. First team data only and then team paired with player data.

One of the objectives in sports analytics is to rate and rank players. In cricket, some possible ranking criteria are statistics such as batting average and strike rate for batsmen in determining most valued players. Lewis, Lemmer, Alsopp and Clarke, and Beaudoin develop new performance measures to rate teams and to find the most valuable players.

WASP is one model which was introduced by Sky Sports of Newzeland in November 2012. It is “winning and score predictor” for use in limited-overs games. WASP is to predict the outcome of two things: first it predicts the score for the first inning based on past records of players and venue secondly, it predicts the outcome of the match based on the past record of the team and venue. The models are based on a database of all non-shortened ODI and 20-20 games played between top-eight countries since late 2006.

In whole literature review, it is found that either they are focusing on historical data or on data from social media but not considering both at a time. Our aim is to gather both types of data at a single point and come up with smart and more effective and more accurate model.

Chapter 3

System Design

3. Introduction:

In order to calculate the results of human emotions we are considering two types of data set to make the results more accurate and reliable. These data sets are: data from past tweets known as Historical Data and data from the real time twitter streaming. Design of these two data sets is explained below.

3.1 Historical Data:

We create machine learning models with a specific end goal to anticipate the results of the human emotions. We utilized a multi-step way to deal with examining the information that delivered **31 features**. Chi-square test were utilized for feature selection. The chose features were utilized as inputs to four distinctive characterization algorithms: naive Bayes, logistic regression, random forests, decision trees and SVM.

Steps involved developing the decision making system (Historical Data) Making the decision from millions of tweets was a difficult task. It involved lots of processes and steps to consider. We categorized all of the flow into six major phases and will discuss them all individually. These 6 steps are given below:

- Data Collection.
- Data Filtration.
- Feature Construction.
- Training & Classification.
- Testing data
- Prediction Outcome.

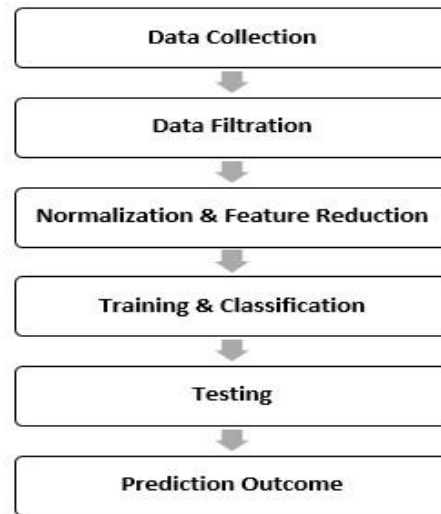


Figure 3.1: Steps involved developing decision making system (Historical Data)

3.1.1 Data Collection:

The first major task is to collect real time data of public media. Data were scraped from the Twitter streaming API <https://apps.twitter.com>. We have collected the data for all over the world.

Twitter Streaming API:

Twitter streaming API give the access of real time data from twitter content. They provide access of large amounts of data from all over the world. Twitter, give tweets, location of tweets, tweet language, tweet sub-continent, tweet country, two-digit language ISO code, and much more depend on your requirements and filtration.

3.1.2 Data Filtration:

The data from Mongo-DB (Historical data) and real time tweets from twitter contains raw data like an ID, date, location, language, screen name and text. We don't need a source of tweet, longitude, latitude, Geo location, background image, foreground image, and personal email. We just need to Summarize tweet that can give us the complete picture of the emotions, and for this purpose we perform data filtration.

3.1.3 Features Construction:

The features are formed within a clear hierarchy where level 1 features are basic performance statistics, level 2 features are combinations of 2 level 1 features and level 3 features are combinations of 2 level 2 features. 31 features were formed in all.

Nine different statistics were used as a starting point for creating the emotive level features and they are outlined in Table 1.

They can be calculated for both the home and away teams, giving 18 base features.

| Emotion Features | Description |
|-----------------------------|--|
| Happy Synset Percentage | $\text{Happy} = (\text{Count-Happy} / \text{Result}) * 100$ |
| Fear Synset Percentage | $\text{Fear} = (\text{Count-Fear} / \text{Result}) * 100$ |
| Joy Synset Percentage | $\text{Joy} = (\text{Count-Joy} / \text{Result}) * 100$ |
| Sadness Synset Percentage | $\text{Sadness} = (\text{Count-Sad} / \text{Result}) * 100$ |
| Language Synset Percentage | $\text{Language} = (\text{Count-Language} / \text{Text}) * 100$ |
| Continent Synset Percentage | $\text{Subcontinent} = (\text{Count-Continent} / \text{Text}) * 100$ |
| Location Synset Percentage | $\text{Location} = (\text{Count-Location} / \text{Text}) * 100$ |
| ISO Language Synset | $\text{ISO-LAN} = (\text{Count-ISO-LAN} / \text{Text}) * 100$ |
| ISO Country Synset | $\text{ISO-CNR} = (\text{Count-ISO-CNR} / \text{Text}) * 100$ |

Table 3.1: Emotive Features

The most important technique for data analytics is to analyze text using string tokenizer for pattern matching (Naïve Bayes Algorithm) and Stanford NLP POS tagger API version 3.6 techniques [4].

Emotion famous words like happy, fear, sadness, and joy put as wordForm one by one to word net 3.0 open source tool to extract synonyms from database of these emotions and store sets separately into datasets [5].

From these base or level 1 features it is possible to calculate more complex features in the following ways.

1. Net Features:

String matching emotion datasets and tweet text extract from file line by line to count the net feature of the emotions. Example, extract tweet text from file line by line and tokenize each string in a line and match using (Naïve Bays Algorithm) with emotion dataset that contains synonyms against each emotion word. Each emotion like happy, and so on counted separately and divided by total tweets then multiply by 100 for a percentage of each emotion. The outcome of each emotion is stored in a Mongo-DB collection as a document.

2. Differentiating Features:

These are the differences in positive tweets, negative tweets and neutral tweets where accurate sentiment analysis of informal genre is important. For example, in a tweet person show his strong opinion with strong words like I am **very** happy with that policy and another site another tweet shows negative impact with strong word like **Not** satisfied with this policy. On the other hand, another tweet doesn't give an accurate word in a tweet for analysis where informal genre is important which shows a neutral situation.

This gives 5 new difference or level 3 features.

These 31 features (18 x level 1, 8 x level 2 & 5 x level 3) were calculated for each pattern in the dataset, using the previous data frame by each emotion to do so.

All Feature Extraction Detail Below

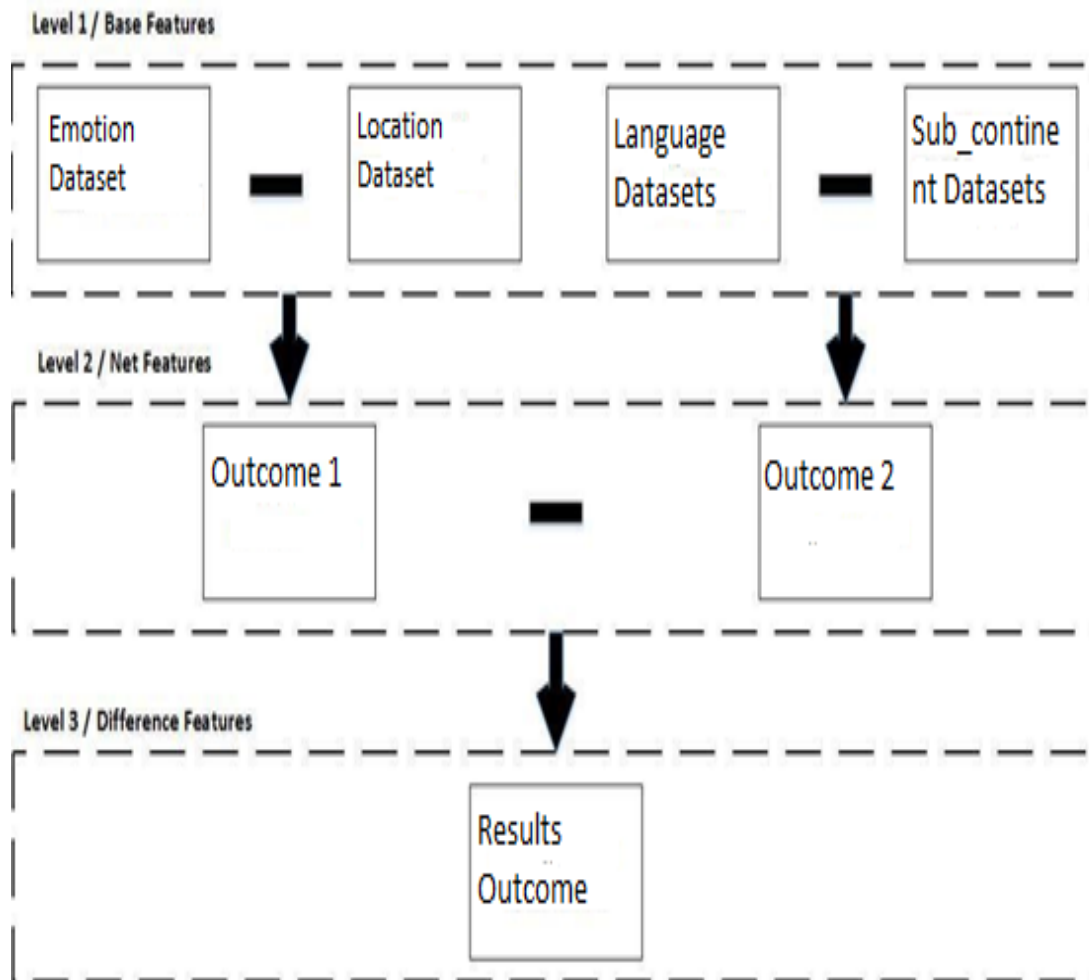


Figure 3.2: Features Construction

List of 31 features is as follows

LEVEL 1 FEATURES

| Sr. No | Team | Feature Name | Formula | Level |
|--------|---------------|-----------------------------|---|---------|
| 1 | Emotion 1 | Happy Synset Percentage | Happy = (Count-Happy / Result) * 100 | Level 1 |
| 2 | Emotion 2 | | | |
| 3 | Emotion 1 | Fear Synset Percentage | Fear = (Count-Fear / Result) * 100 | |
| 4 | Emotion 2 | | | |
| 5 | Language 1 | Language Synset Percentage | Language = (Count-Language / Text) * 100 | |
| 6 | Language 2 | | | |
| 7 | Continent 1 | Continent Synset Percentage | Subcontinent = (Count-Continent / Text) * 100 | |
| 8 | Continent 2 | | | |
| 9 | Location 1 | Location Synset Percentage | Location = (Count-Location / Text) * 100 | |
| 10 | Location 2 | | | |
| 11 | ISO Lang 1 | ISO Language Synset | ISO-LAN = (Count-ISO-LAN / Text) * 100 | |
| 12 | ISO Lang 2 | | | |
| 13 | Emotion 1 | Sadness Synset Percentage | Sadness = (Count-Sad / Result) * 100 | |
| 14 | Emotion 2 | | | |
| 15 | Emotion 1 | Joy Synset Percentage | Joy= (Count-Joy / Result) * 100 | |
| 16 | Emotion 2 | | | |
| 17 | ISO Country 1 | ISO Country Synset | ISO-CNR = (Count-ISO-CNR / Text) * 100 | |
| 18 | ISO Country 2 | | | |

Table 3.2: Level 1 Features

LEVEL 2 FEATURES

| Sr. No | Team | Feature Name | Formula | Level |
|--------|-----------|------------------|--------------------------------------|---------|
| 1 | Emotion 1 | Happy Percentage | Happy = (Count-Happy / Result) * 100 | Level 2 |
| 2 | Emotion 2 | | | |
| 3 | Emotion 1 | Fear Percentage | Fear = (Count-Fear / Result) * 100 | |
| 4 | Emotion 2 | | | |
| 5 | Emotion 1 | Joy Percentage | Joy= (Count-Joy / Result) * 100 | |
| 6 | Emotion 2 | | | |
| 7 | Emotion 1 | Sad Percentage | Sadness = (Count-Sad / Result) * 100 | |
| 8 | Emotion 2 | | | |

Table 3.3: Level 2 Features

LEVEL 3 FEATURES

| Sr. No | Feature Name | Formula | Level |
|---------------|----------------------------------|---|----------------|
| 1 | Emotion Difference | Emotion 1 Happy Percentage – Emotion 2 Fear Percentage | Level 3 |
| 2 | Language Difference | Language 1 English Percentage – Language 2 Urdu Percentage | |
| 3 | Sub-Continent Difference | Subcontinent 1 Asia Percentage – Subcontinent 2 Oceania Percentage | |
| 4 | Location Difference | Location 1 Pakistan Percentage – Location 2 United States Percentage | |
| 5 | Outcome Percentage Difference | Result Outcome 1 Net Percentage – Result Outcome 2 Net Percentage | |

Table 3.4: Level 3 Features

3.1.4 Training and Classification:

Machine Learning deals with program learning from data sets. Training data is the data on which the machine learning programs learn to perform correlation tasks. For training and classification, we use 70% of our data for training. Use multiple models of that data so that we can improve our results outcome.

3.1.5 Testing Data:

Testing data is the data, whose outcome is already known (even the outcome of training data is known) and is used to determine the accuracy of the machine learning algorithm, based on the training data. We use 30% of our data set to check the accuracy of our results.

3.1.6 Results, Outcome:

With a specific end goal to make outcome we simply need to give tweet text along with a dataset of emotions like happy, fear, sadness, joy and count results using built in spark engine **word count class**. Analyze these results and store into Mongo-DB as a document form. Final results store in JSON file and put into the Elastic search for more accuracy of results and show these results visualization on Kibana in the form of Line Chart, Pie Chart, Vertical Bar Chart, Histogram. This is further explained in the next section.

3.2 Social Media Data:

Twitter is a popular micro-blogging service, launched in 2006 with more than 25 million unique monthly visitors, where users create status messages (called “**tweets**”). These tweets sometimes express opinions about different topics with a maximum length of 140 characters, which is visible on the public display. The public timeline conveying the tweets of all users worldwide is an

Extensive real-time information stream of more than one million messages **per hour**. Especially during any events like sports match cricket, users express their great interest in the favor of their team or the team to which they are supporting. Word Net API implemented a method to automatically extract sentiment (**Synset**) from a tweet and calculate the results of the **Emotions** according to the expressions of the users. There has been a large amount of research in the area of sentiment classification. The implemented system can produce comparable results on tweets with distant supervision.

In order to train a classifier, supervised learning usually requires hand-labeled training data. With the large range of topics discussed on Twitter, it would be very difficult to manually collect enough data to train a sentiment classifier for tweets.

3.2.1 Characteristics of Tweets:

Twitter messages have many attributes, which uniquely identifies our work among the field of machine learning:

□ Length

The maximum length of a Twitter message is 140 characters. From studies, it has been calculated that the average length of a tweet is 14 words or 78 characters. This is very different from the previous sentiment classification research that focused on classifying longer bodies of work, such as movie reviews.

□ **Data availability**

Another difference is the magnitude of data available. With the Twitter API, it is very easy to collect millions of tweets for training. In past research, tests only consisted of thousands of training items.

□ **Language model**

Twitter users post messages from many different media, including their cell phones. The frequency of misspellings and slang in tweets is much higher than in other domains.

□ **Domain**

Twitter users post short messages about a variety of topics unlike other sites which are tailored to a specific topic. This differs from a large percentage of past research, which focused on specific domains such as movie reviews.

Steps involved developing the decision making system via twitter

Making the decision from millions of tweets was a difficult task. It involved lots of processes and steps to consider. We categorized all of the flow into six major phases and will discuss them all individually. These 6 steps are given below:

- Data Collection.
- Data Filtration.
- Normalization & Feature Reduction.
- Training & Classification.
- Testing Data
- Prediction Outcome.

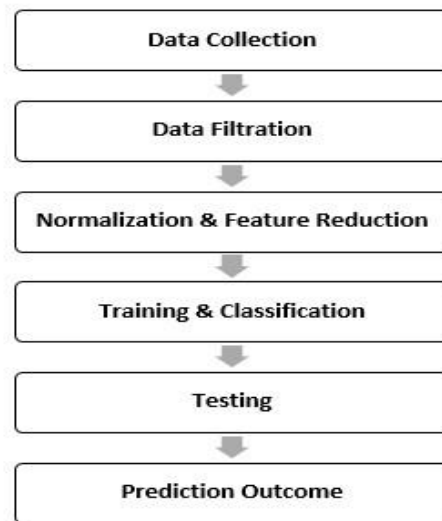


Figure 3.3: Steps involved developing decision making system (Twitter Data)

3.2.2 Data Collection:

The first challenge was to collect the right data. We applied multiple queries to fetch huge data from twitter. Once we got the data, we extracted the selected attributes that helped us in data filtering step. All the other information regarding each tweet was discarded and the extracted data was saved in the Database for the purpose of using in the next phases of the model generation. The whole process was completed in several steps. For data extraction, we have to perform some queries. These queries are then passed to twitter API to fetch data from twitter. All the useful information is obtained.

Then this information or data are placed in a database for permanent storage.

3.2.3 Data Filtering:

After obtaining the data, we filtered the tweets from spam user or spam content. We retrieved the stored data from database and passed it through a series of filtration steps. We considered several factors that classify the tweets from spam or ham. Like some posts ask the user to follow in order

To win prizes, removing the content that is too short to express some meaningful information. Some people use too many hash tags so that their post can reach as many people as possible, therefore we have to remove the tweets having a number of hash tags. We have to consider the tweets of those people who tweets on regular bases there are many users who hardly post tweets so we are not going to consider those tweets. We are also neglecting the user with very few followers that means these users don't have any influence on other people, or maybe they are not a regular user of twitter.

3.2.4 Normalization and Feature Reduction:

Structured Tweets are generally in sentence format, with URLs specified for images or blog articles. To get data that are in usable format we remove the stop words that contains general terms like a, the, etc. and emoticons.

3.2.5 Training and Classification:

Machine Learning, as you may know, deals with program learning from data sets. Training data is the data on which the machine learning programs learn to perform correlation tasks. For training and classification, we use 70% of our data for training.

Use multiple models of that data so that we can improve our prediction.

3.2.6 Testing Data

Testing data is the data, whose outcome is already known (even the outcome of training data is known) and is used to determine the accuracy of the machine learning algorithm, based on the training data. We use 30% of our data set to check the accuracy of our results.

3.2.7 Prediction Outcome

In order to make predictions we just have to give the name of the emotion along with an opponent's name and the system will give the calculation of the emotions after calculating different task. This is further explained in the next chapter.

Chapter 4

System Implementation

4. Introduction:

Implementation for past tweets and real time tweets from social media data are given below

4.1 Implementation of Historical Data:

4.1.1 Data Collection:

Data is collected from Twitter streaming API. The twitter API is provided at a real time streaming, one of which contains all of the tweets detail and expressions, and the others certain sub-sets of tweets location, such as for type of tweets, tweet subcontinent in certain countries, tweet ID, or genders and periods of time.

Data Format

The data are provided in [YAML](#) (Yet Another Markup Language) format, a human-readable data format. There are libraries available to parse this in multiple languages. As for the structure of the file it is clear enough when you have a look at the data.

4.1.2 Data Filtration:

As we have mentioned above twitter streaming API provides real time data streaming for every tweet, and we don't want to have all information about data in a tweet form, for this we have to summarize this data. In order to summarize data, we have used Princeton University API called WordNet library.

WordNet

This Princeton university package can be used to analyze the performances of tweets based on word **synset** data from Twitter Streaming API [5]. The WordNet package can handle word data from twitter server. The data have to be in yaml format. Using this **API**, we make processing on that yaml data to create a database for all Emotions.

The database consists of following data:

Tweet, Tweet ID, Tweet Date, Tweet location, Tweet Source, Tweet screen name, Tweet language, Tweet Subcontinent, Two-digit language ISO code.

4.1.3 Feature Construction

After applying WordNet API for data, we retrieved the stored data from Mongo dB using apache spark for feature construction. Then 31 features are created that list down in chapter 3. These features are then saved in a JSON file.

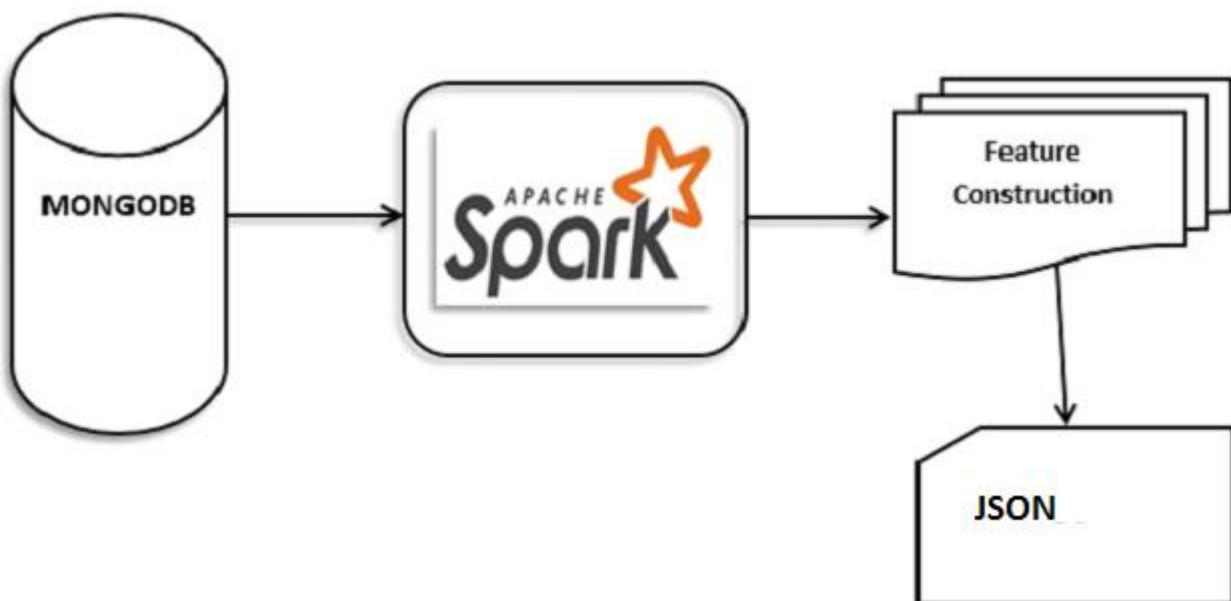


Figure 4.1: Features Construction

4.1.4 Training and Classification

Machine Learning deals with program learning from data sets. Training data is the data on which the machine learning programs learn to perform correlation tasks. For training and classification,

We use 70% of our data for training. Use multiple models of that data so that we can improve our prediction. We have used five different models for training our data set.

Naive Bayes, Logistic Regression, Random Forests, Decision Trees and SVM.

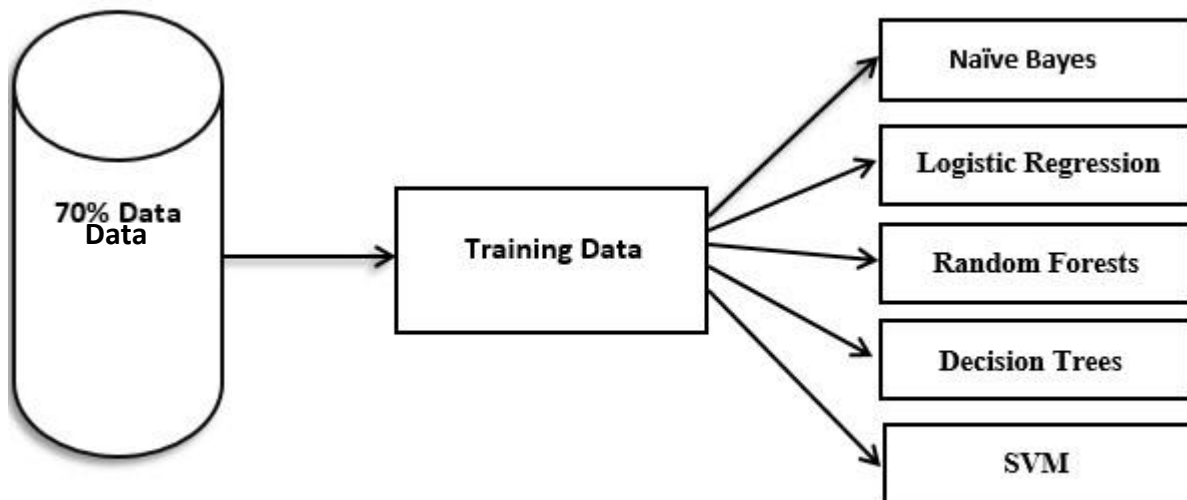


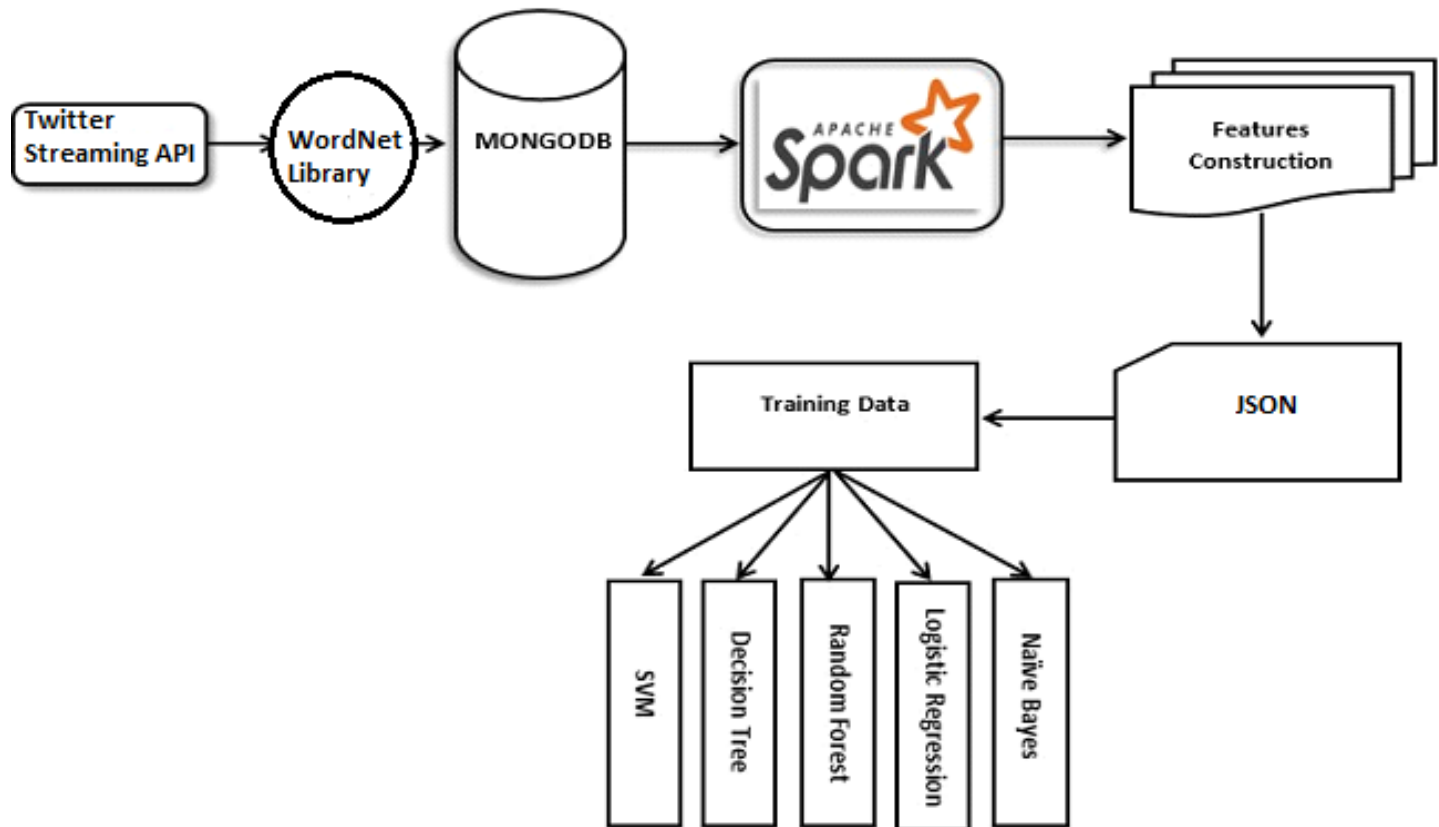
Figure 4.2: Training and Classification

4.1.5 Testing Data

Testing data is the data, whose outcome is already known (even the outcome of training data is known) and is used to determine the accuracy of the machine learning algorithm, based on the training data. We use 30% of our data set to check the accuracy of our results. This 30% data were randomly tested with all above mentioned models to test the accuracy. Using Naïve Bayes, we got a maximum of accuracy that is 68% approximately.

4.1.6 Prediction outcome

In order to make predictions we just have to give the name of emotions and opponent name, the system gets the features from historical data and also from real time data for these four emotions analyzes the record and calculate the outcome of the emotions.

Block Diagram (Historical Data)**Figure 4.3: Block Diagram (Historical Data)**

4.2 Implementation of Social Media Data

4.2.1 Data Collection

The first challenge was to collect the right data. We applied multiple queries to fetch huge data from twitter. Once we got the data, we extracted the selected attributes that helped us in data filtering step. All the other information regarding each tweet was discarded and the extracted data was saved in the Database for the purpose to be used in the next phases of the model generation.

The whole process was completed in 5 steps explained below

a. Queries

In twitter, we have multiple queries referring to a specific topic or event. We used a group of query strings to gather a large number of tweets for the better training of our model. During the process of implementation, we used 6 query strings referring to a single topic.

b. Twitter API

Every single query was then passed to the Twitter API to fetch data. At least 1000 tweets against each query were retrieved and were passed to the next step of the data collection process. At the time of implementation, we acquired **26,777** tweets.

c. Results

All of these tweets were added into an **array list**. Each tweet contained all of the raw information about the tweet and the user posting that tweet.

d. Data extraction

The raw information was filtered and only the required information was obtained from the above results. We gathered **Status ID, Source, Text, User Status Count** and **User Follower Count**. All of the other information was discarded for better efficiency and space consumption.

e. Mongo dB

All of the extracted info was then moved from **RAM** to permanent storage i-e. Mongo dB, so that it can be used for further processing in the upcoming phases of model generation and prediction.

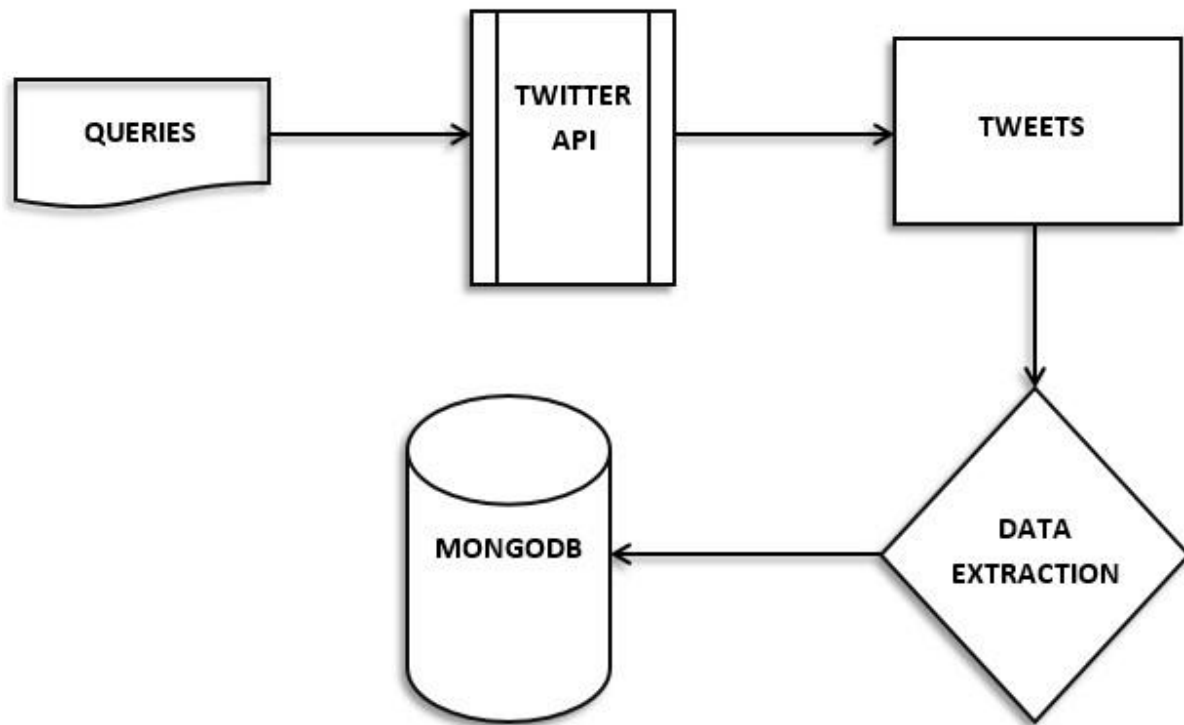


Figure 4.4: Data Collection (Social Media)

4.2.2 Data Filtering

After obtaining the data, we filtered the tweets from spam user or spam content. We retrieved the stored data from Mongo dB using apache spark and passed it through a series of filtration steps. We considered 6 factors that classify the tweets from spam or ham. Those factors are explained below.

1. Content is requesting re-tweets and follows

Spam created by competitions often asks users to retweet posts or to follow profiles to win prizes.

You can filter this out simply **string contains method**.

```
// Remove content requesting follows or retweets
```

```
return spamwords.contains(tweetword) ? true : false
```

2. Short content length

Often users will write very short posts, such as '@friend ok' as a response to a question.

This content has little value in analysis.

```
// Remove content with less than 10 characters
```

```
return tweet.length() > 10? true : false
```

Results: After successfully applying first 2 steps, we obtained **25,303** tweets from a total of **26,777**, which means that **1300** tweets were considered to be as spam and were discarded.

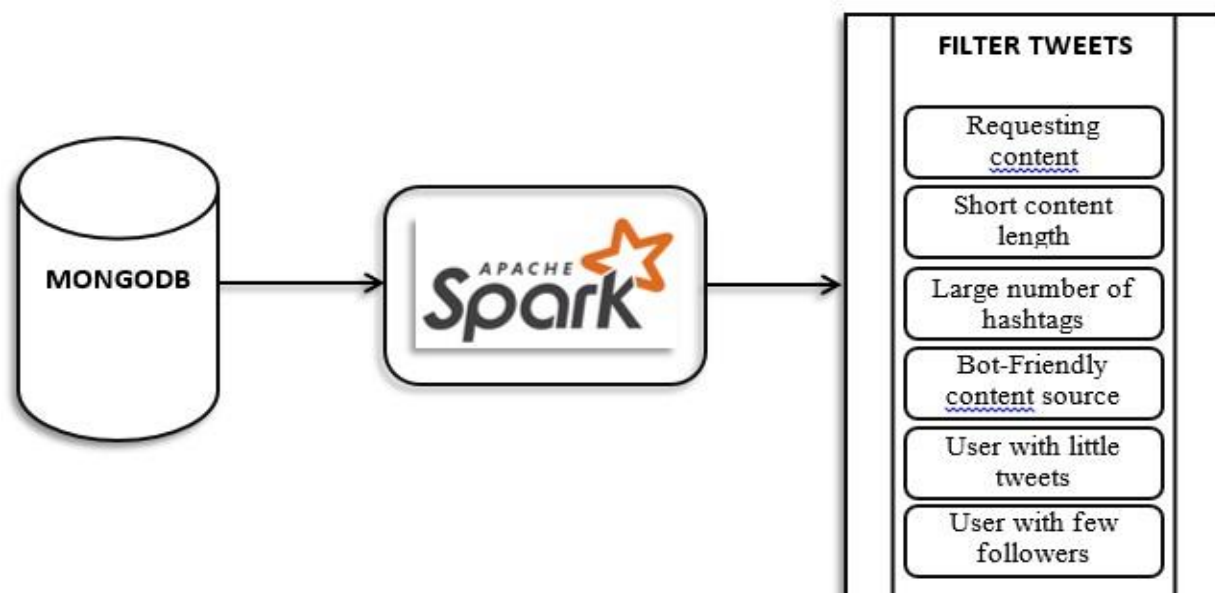


Figure 4.5: Data Filtration (Social Media)

3. Large numbers of hashtags

Poor quality content tends to include many hashtags. Many hashtags might be used by spam creators to hope that they can reach as many users listening to those tags as possible.

```
// Remove content with more than 5 hashtags
```

```
return word.startsWith("#")? true : false
```

Results: After applying this step, we obtained **25,223** tweets from a total of **25,303** and the remaining tweets were discarded.

4. Bot-Friendly content source

Twitter provided with the facility to fetch the source, target that exposes the application or service used to post content. Some services have a reputation for being 'friendly' to bots.

```
// Remove content from bot-friendly services
```

```
Status.getUser().getSorce()
```

Results: After applying this step, we obtained **19,855** tweets from a total of **25,223** and the remaining **5,368** tweets were considered as spam and were ignored.

5. Users that create little content

If a user has hardly ever tweeted it may be that the user was created by a bot to deliver a small number of tweets before it is discarded.

```
// Remove users who have only created less than 50 tweets
```

```
Status.getUser().getStatusesCount()
```

Results: After applying this step, we obtained **19,097** tweets from a total of **19,855** and the remaining tweets were ignored.

6. Users with few followers

If a Twitter profile is created just to post spam messages, the profile is likely to follow lots of users, but be followed by very few users itself. Users with less than 50 followers are discarded.

```
// Remove content from users with few followers
```

```
Status.getUser().getFollowersCount()
```

Results: After applying this step, we obtained **17,308** tweets from a total of **19,097** and the remaining tweets were ignored.

4.2.3 Normalization and Feature Reduction

Structured Tweets are generally in sentence format, with URLs specified for images or blog articles. To get data that are in usable format we remove the stop words that contains general terms like a, the, etc. and non-English words. We performed following operations on tweets in cleansing and normalizing phase.

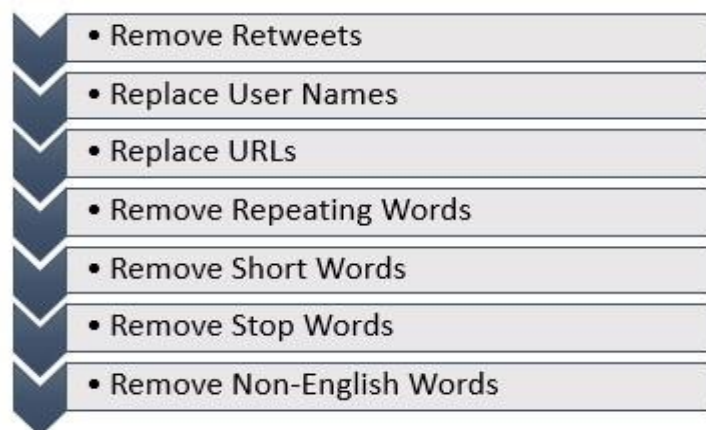


Figure 4.6: Normalization and Features Reduction (Social Media)

1. Remove Retweets

Retweeting is the process of copying another user's tweet and posting to another account. This usually happens if a user likes another user's tweet. Retweets are commonly abbreviated with “**RT**”. For example, consider the following tweet: *Awe-some! RT @rupertgrintnet Harry Potter Marks Place in Film History <http://bit.ly/Eusxi> :)*. In this case, the user is rebroadcasting rupertgrintnet's tweet and adding the comment *Awesome!* Any tweet with RT is removed from the training data to avoid giving a particular tweet extra weight in the training data.

2. Replace Usernames

Users often include Twitter usernames in their tweets in order to direct their messages. A de facto standard is to include the @ symbol before the username (e.g. @daniyalbutt99). We replaced all the usernames with the keyword “**USER_NAME**”.

3. Replace URLs

Users very often include links in their tweets. All the URLs like “<http://tinyurl.com/cvvg9a>” are replaced with the keyword “**URL**”.

4. Remove Repeated letters

Tweets contain very casual language. For example, if you search “wow” with an arbitrary number of o's in the middle (e.g. woooow, woouooooooooow, woouooooooooooooow) on Twitter, there will most likely be a non-empty result set. We use preprocessing, so that any letter occurring more than two times in a row is replaced with two occurrences. In the samples above, these words would be converted into the token *woow*.

5. Remove Short Words

Sentence may contain very short words, such as 'ok' or 'at' that cannot be considered as a feature. These words have little value in analysis. Therefore, we removed any word with less than 3 characters.

6. Remove Stop Words

Words that don't have any value in the analysis process are considered as stop words. e.g. 'being', 'there'. They just increase the number of features when classified. We removed these words to reduce the features for better results.

7. Remove Non English Words

Twitter is a multilingual website; it contains tweets from the different regions of the world with different languages. For the prediction purpose, we only considered English tweets and removed non-English words from a tweet. This task is accomplished with the use of regex pattern.

4.2.4 Machine Learning Methods

We test 2 classifiers: keyword polarity-based and Naive Bayes for the prediction of the emotion data analytics result.

4.2.4.1 Baseline

We initially applied polarity based classification methods using the set of positive and negative words provided by **AFINN**, which is a list of **2477** English words rated for valence with an integer between minus five (negative) and plus five (positive). As a baseline, we use AFINN's list of keywords. For each tweet, we count the number of negative keywords and positive keywords that appear. This classifier returns the polarity with the higher count. After acquiring the polarity of the

tweets, we saved the tweets along with its polarity into a JSON file that will be used in the model generation process.

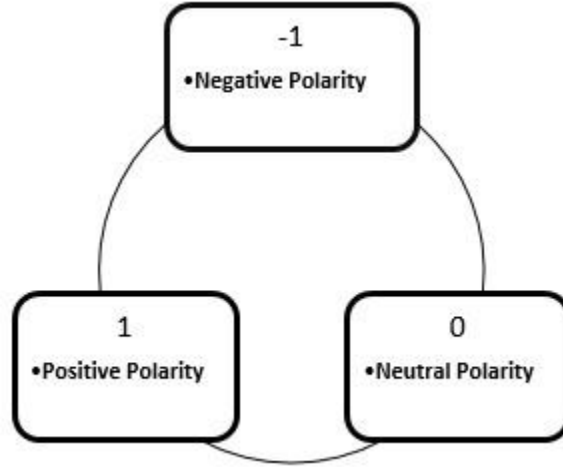


Figure 4.7: Polarity Base Classification

4.2.4.2 Naïve Bayes

Naive Bayes is a simple model which works well on text categorization. We use a multinomial Naïve Bayes model. Class c^* is assigned to tweet d ,

where $c^* = \operatorname{argmax}_c P_{NB}(c/d)$

$$P_{NB}(c/d) := \frac{(P(c) \sum_{i=1}^m P(f_i|c)^{n_i(d)})}{P(d)}$$

In this formula, f represents a feature and $n_i(d)$ represents the count of feature f_i found in tweet d .

There is a total of m features. Parameters $P(c)$ and $P(f/c)$ are obtained through maximum likelihood estimates, and add-1 smoothing is utilized for unseen features.

Model Generation Process Diagram

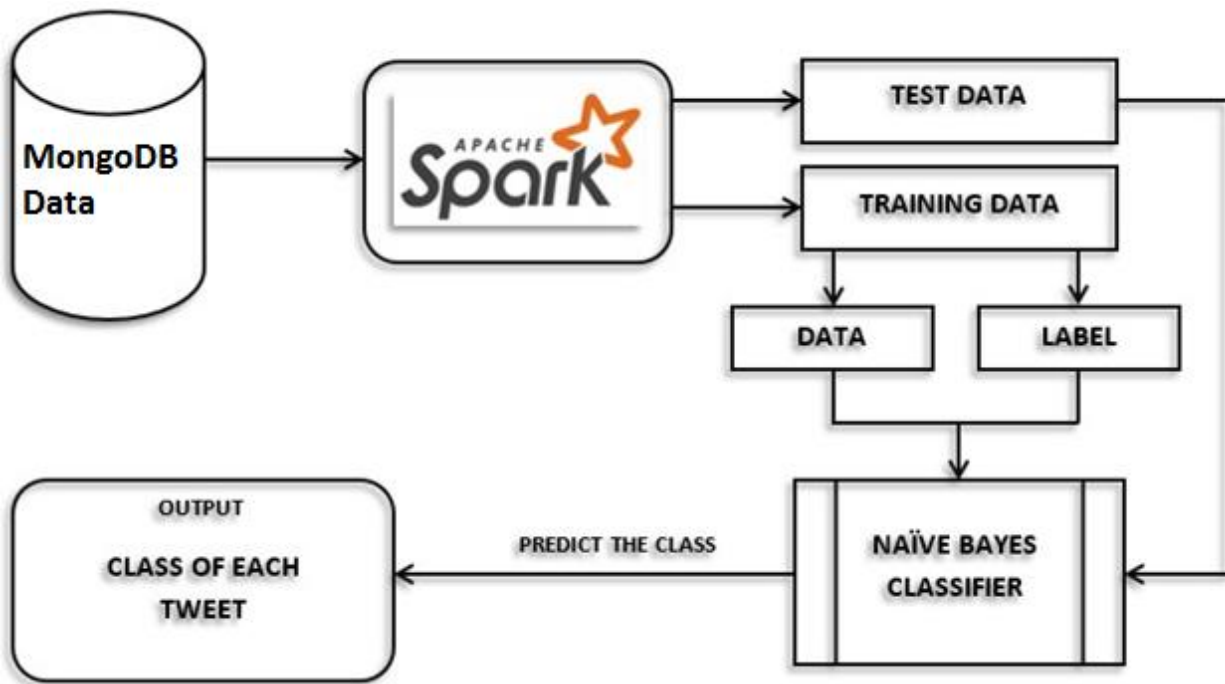


Figure 4.8: Naïve Bayes Model Generation Process

Steps

- We imported the saved Mongo data into apache spark and converted the data into RDD (Resilient distributed dataset) of labeled point (Java RDD < Labeled Point >). A labeled point is a local vector, dense or sparse, associated with a label/response. In mllib, labeled points are used in supervised learning algorithms. We used a double to store a label, so we could use labeled points in both regression and classification. For binary classification, a label should be either 0 (negative) or 1 (positive). For multiclass classification, labels should be class indices starting from zero: 0, 1, 2 ...

- After converting the data into labeled Points, we randomly split the data into two; training and test data. We kept the ratio of 70: 30 i-e 70% of the total data was taken for training purpose, while 30% of the total data were engaged in testing.
- After that, we applied Naïve Bayes Model on our training data using apache spark mllib. Once we obtained our model ready, the test data were applied on our model for testing the accuracy of our model.

4.3 Prediction through Streaming Tweets

Twitter open sourced its **Hosebird client** (hbc), a robust Java HTTP library for consuming Twitter's Streaming API. We used *hbc* to create a Kafka twitter stream producer, which tracked our query terms in twitter statuses and produced a Kafka stream out of it, which was utilized later for sending that data from Kafka to Spark Streaming.

Once Kafka producer started working, we retrieved the produced messages from spark streaming and applied the same filtration and normalization steps. After successfully passing through all of the steps, the tweets were passed to the naïve Bayes model that we generated before for prediction purpose. The system then performed its computation and gave us the polarity of the tweet that was then processed to predict the calculation of the emotions according to the thinking of the twitter users. Mainly used technologies to achieve all of the above flow are defined below:

4.3.1 Spark Streaming

Spark Streaming is a real-time processing tool that runs on top of the Spark engine.

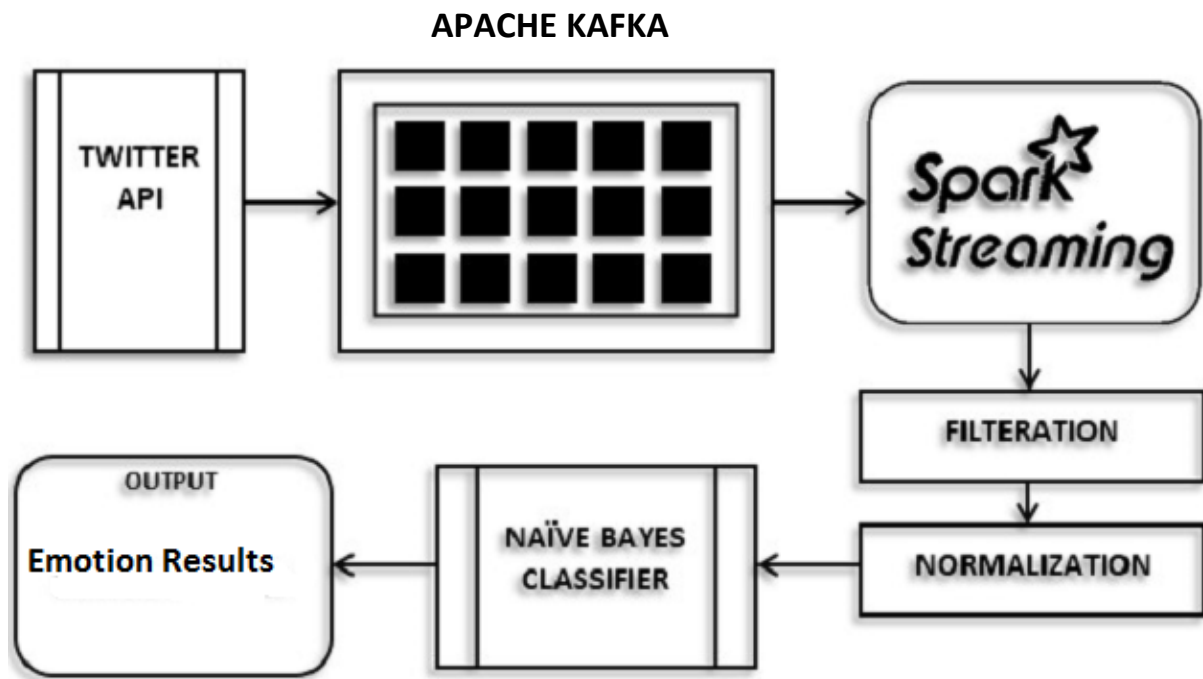


Figure 4.9: Prediction Process

a. Apache Zookeeper

Apache Zookeeper is an effort to develop and maintain an open-source server which enables highly reliable distributed coordination.

b. Apache Kafka

Apache Kafka is publish-subscribing messaging rethought as a distributed commit log.

Steps to create Kafka Producer

Following are the important steps that we used to create kafka producer for fetching twitter data:

1. Set the properties to configure Kafka Producer to publish messages to a topic


```
Properties properties = new Properties();  
properties.put("metadata.broker.list", "localhost:9092");  
properties.put("serializer.class", "kafka.serializer.StringEncoder");  
properties.put("client.id", "tweet");
```

2. Next we set up a *StatusFilterEndpoint* , which will setup track terms to be tracked on recent status messages, as in the example, *twitterapi*

```
StatusesFilterEndpoint endpoint = new StatusesFilterEndpoint();  
endpoint.trackTerms(Lists.newArrayList("twitterapi"));
```

3. After that we provided authentication parameters for OAuth for using twitter.

```
Authentication auth = new OAuth1(cred[0], cred[1], cred[2], cred[3]); client  
= new ClientBuilder().hosts(Constants.STREAM_HOST)  
.endpoint(endpoint).authentication(auth)  
.processor(new StringDelimitedProcessor(queue)).build();
```

4. Last step, connect to the client, fetch messages from the queue and send through Kafka Producer

```

client.connect();

for (int msgRead = 0; msgRead < 1000; msgRead++) {

    KeyedMessage<String, String> message = null;

    try {

        String msg = queue.take();

        message = new KeyedMessage<String, String>(topic, msg);

    } catch (InterruptedException e) {

        e.printStackTrace();

    }

    producer.send(message);

}

```

Block Diagram Social Media

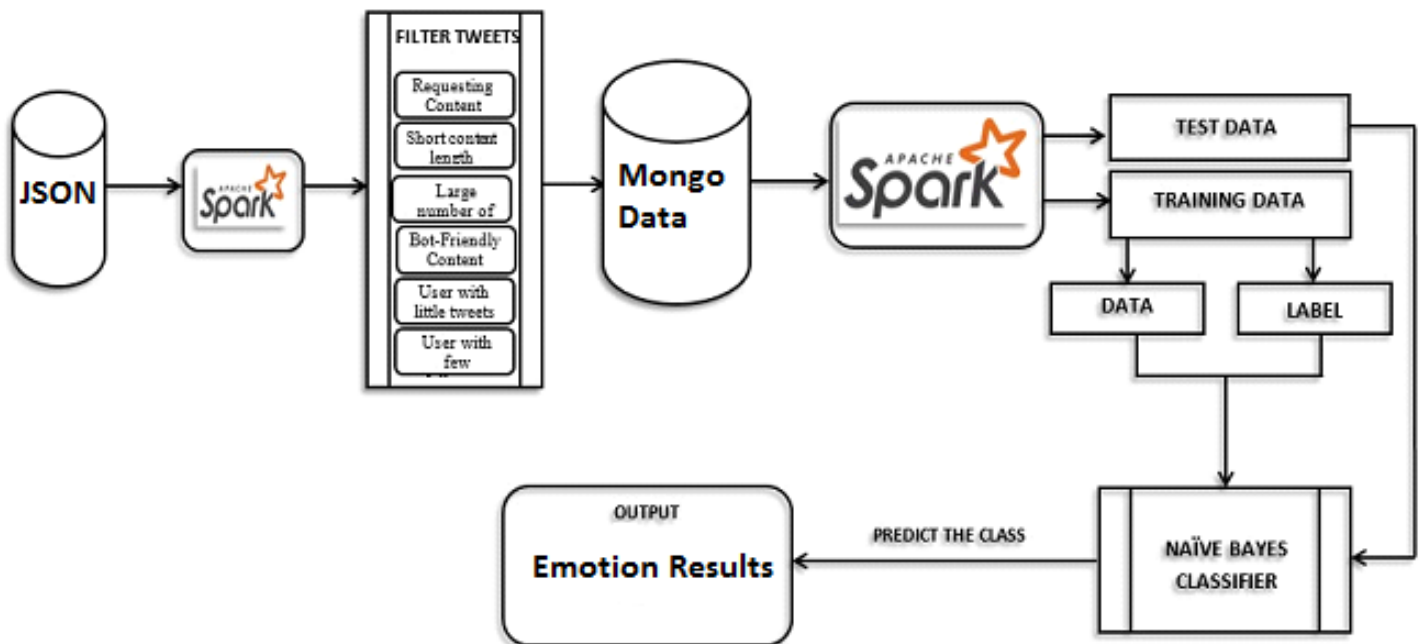
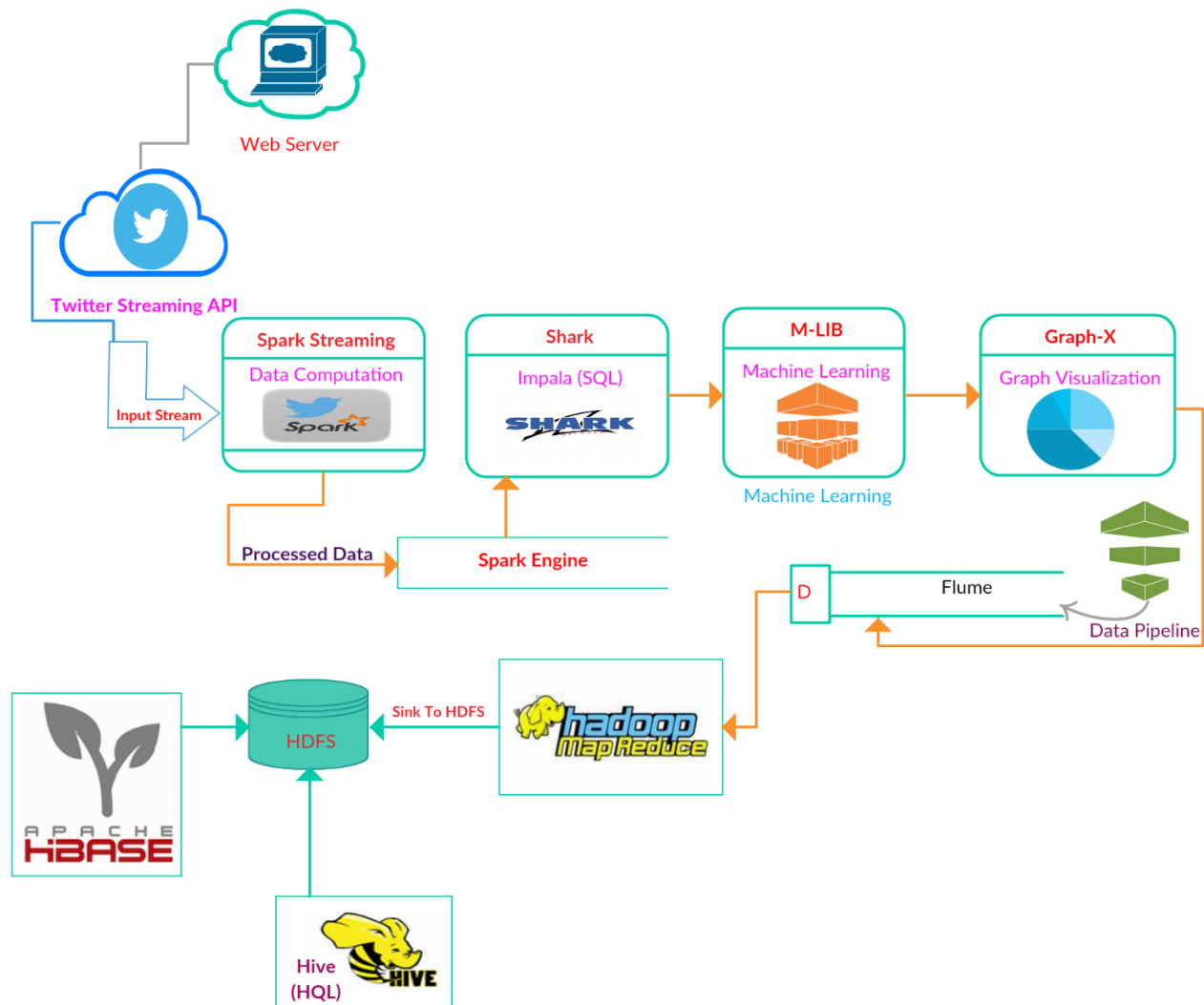


Figure 4.10: Block Diagram Social Media

Data Flow Process:

online diagramming & design | creately.com

4.3.2 Apache Spark

Apache spark provides lightning-fast cluster computing. Apache spark is an open source general large scale distributed computational engine. Now a day's **spark** is one of the most widely used processing engine for Big data analytics. Spark has a collection of rich libraries and language integrated API's.

Spark provides support of these three big languages, **Python, Java, Scala, and also R language** where Scala is most supportable and very fast for big data analysis data computation in spark. Java and python also have a large number of built in libraries for data computation in spark [6].

The Spark is 100x faster than **Hadoop** and spark run on top of Hadoop using HDFS (Hadoop Distributed File System). Spark's work on **Map reduce** jobs, which are the first used in (Google File System) and after adopting this technique in Apache Hadoop. Apache spark is fast because it processes data in memory for data computation. As Spark transitioned from early adopters to a broader audience, we had a chance to see where it functional API worked well in practice, where it could be improved, and what the needs of new users were.

4.3.3 Apache Spark Core Libraries

Spark core libraries detail in figure 4.11 below

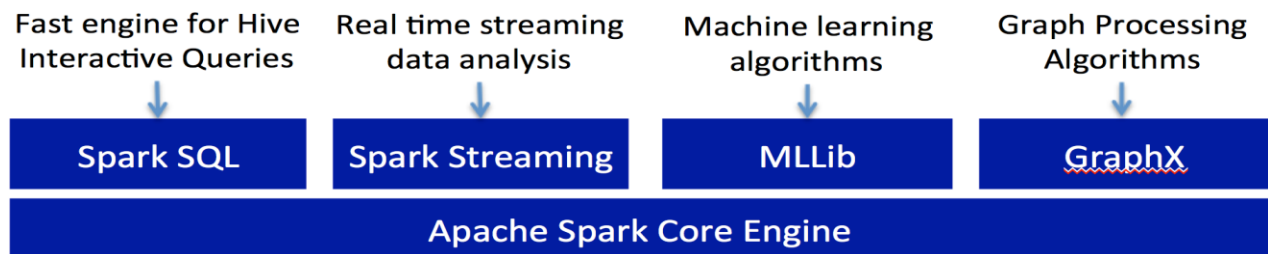


Figure 4.11: Spark Core Libraries Detail

4.3.4 Apache Spark Processing on Data UI

Batch processing is performed on twitter streaming to collect data from server in a multiple stream.

Batch processing **Scheduling** the multiple streams that having approximately. Delay every batch

Processing Time is 2 seconds to perform operations and deleting the batches to optimize the

space. **Total Delay** is the time of twitter server for Mongo DB, delay that occur on processing of data.

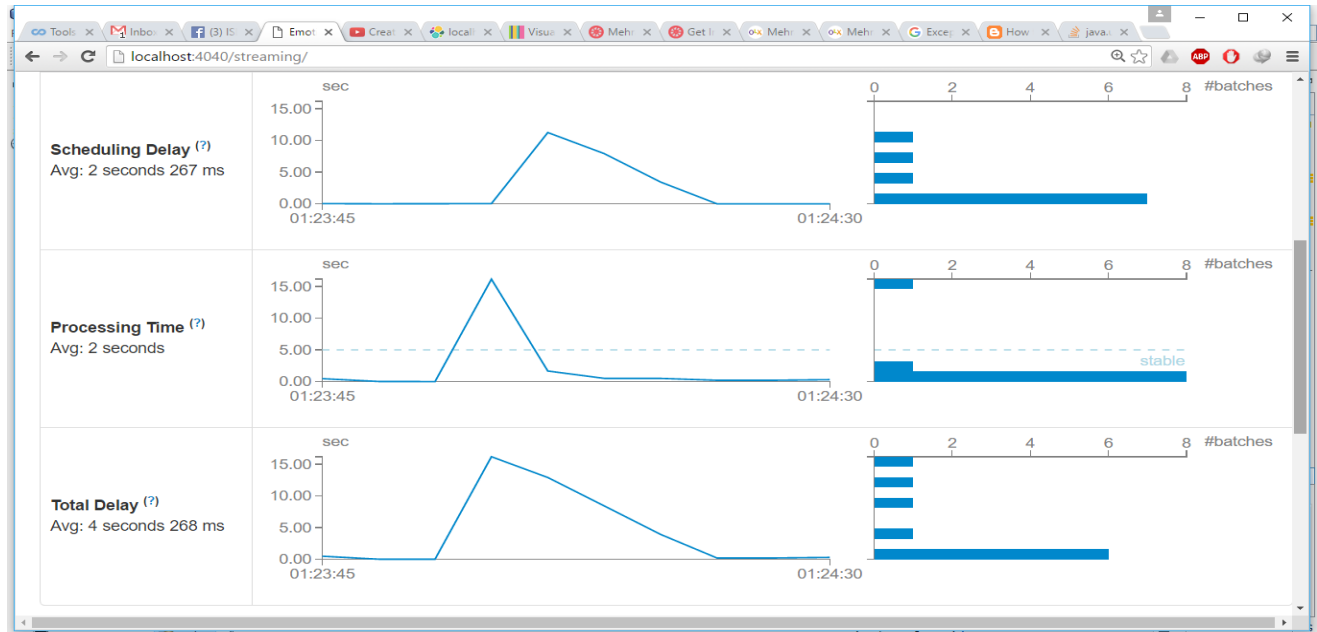


Figure 4.12: Spark Scheduling Delay, Processing Time, Total delay

4.3.5 Apache Spark Processing on Spark Job

Total spark **running time** 2.8 min and scheduling mode considered FIFO (First in First Out) to handle Multiple batches of data differentiated through specific ID and after completion, the batches are deleted. All batches are marked as blue color and green color denoted total running time from 45 to 0.

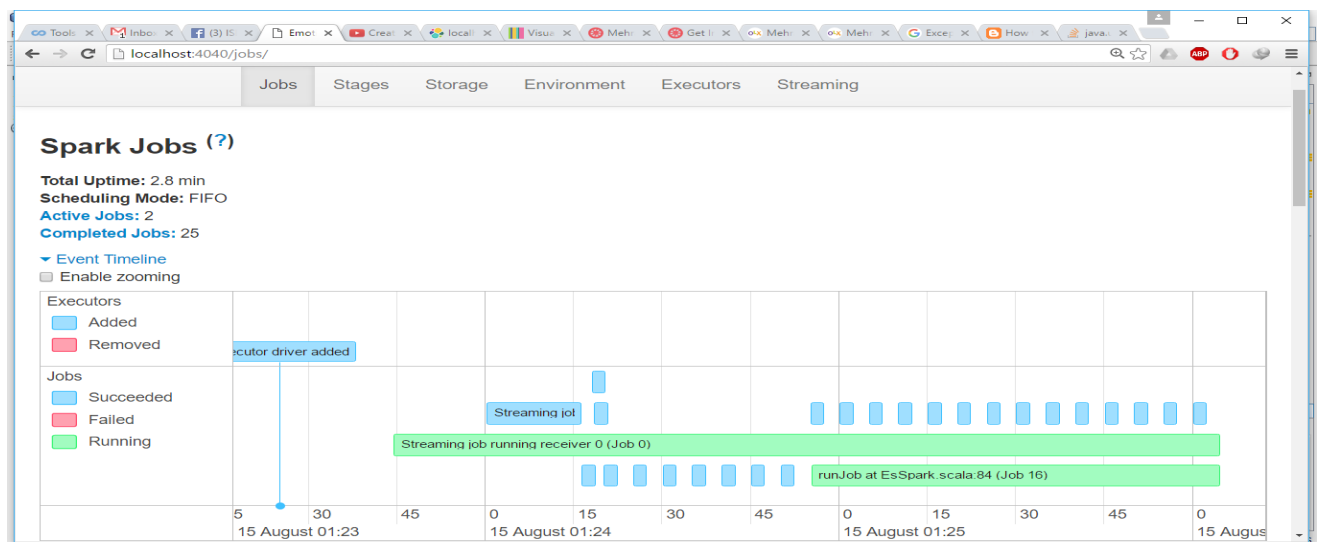


Figure 4.13: Spark Jobs

4.3.6 Apache Spark Processing on Spark Executors

Total **RAM** memory used **438.0 MB** and disk **0B**. It's used to handle multiple clusters for **RDD** Processing. Input size in one millisecond is **674 B**. Total tasks are **413**. Completed tasks are **411** and remaining **two task** are still running and collecting data as input [7].

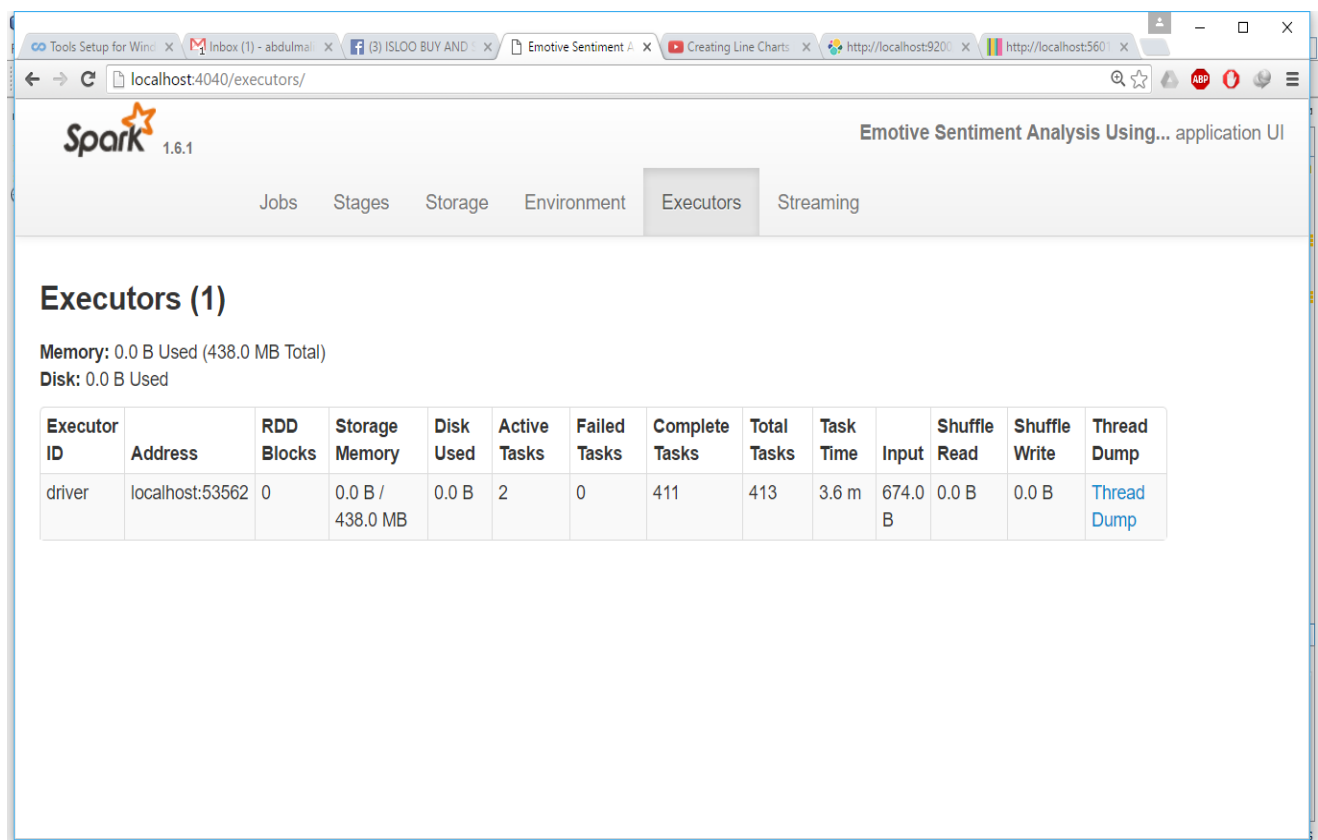


Figure 4.14: Spark Executors

4.3.7 Apache Spark Streaming with Twitter

Batch processing is performed on twitter streaming to collect data from server in a multiple stream.

Input rate used to extract average events per second from the twitter stream. Batch processing

Scheduling the multiple streams that having five minutes 4 second delay. Delay every batch

Processing Time is 3 seconds to perform operations and deleting the batches to optimize the

space. **Total Delay** is the time of twitter server for Mongo Db delay that occur on processing of data.

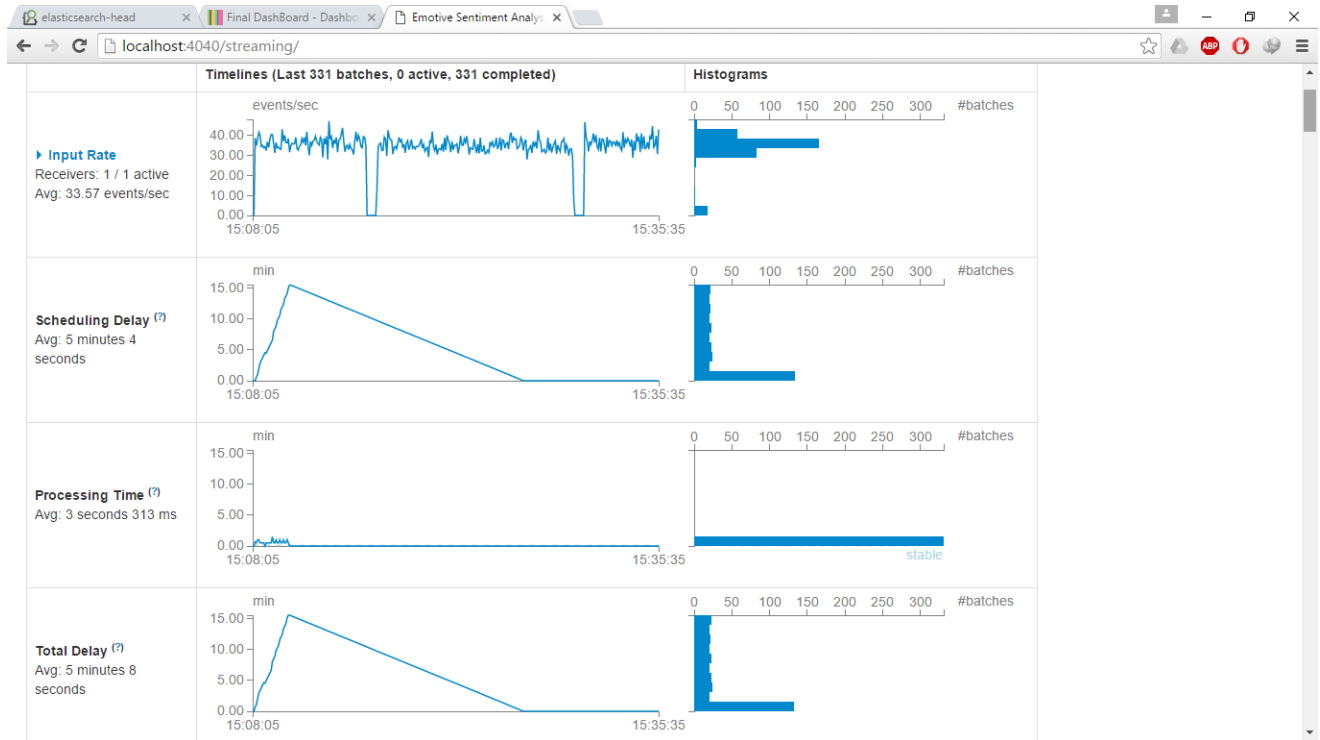


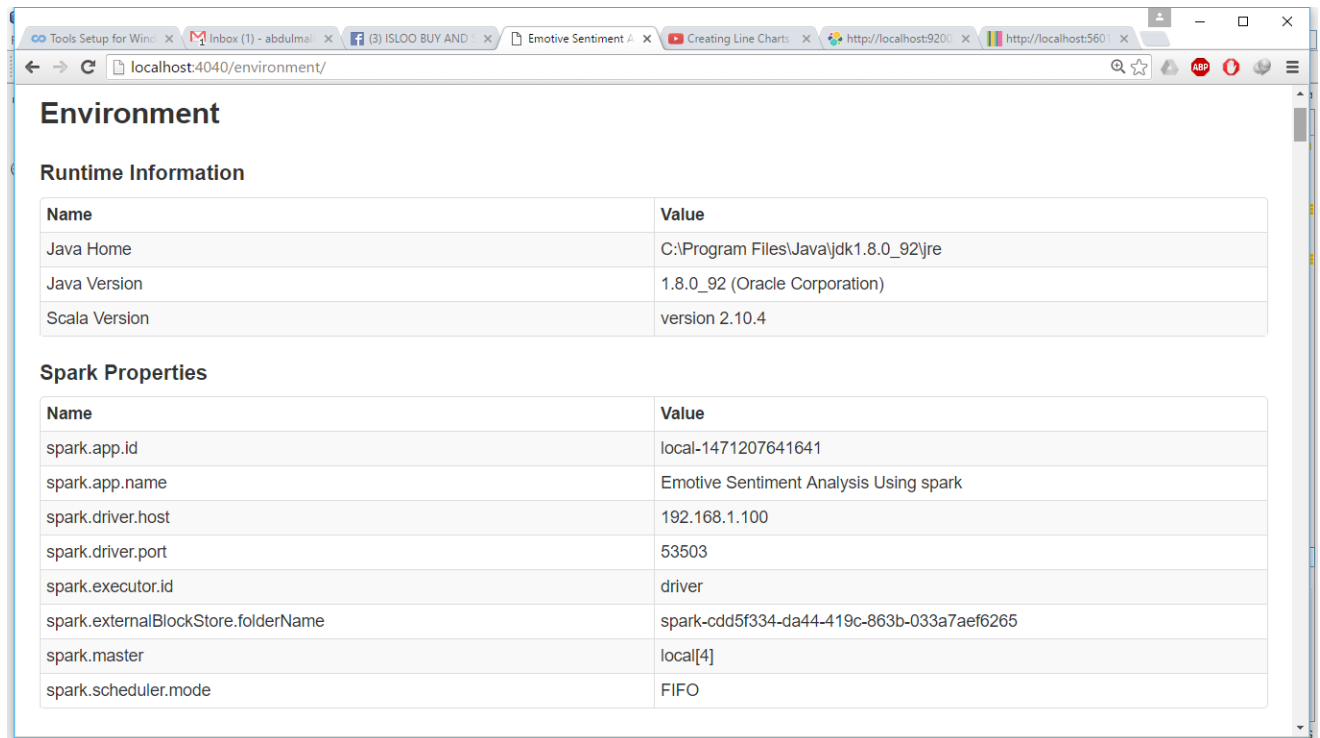
Figure 4.15: Spark Streaming with Twitter

4.3.8 Apache Spark Processing Completed Jobs

| Job Id | Description | Submitted | Duration | Stages: Succeeded/Total | Tasks (for all stages): Succeeded/Total |
|--------|---|---------------------|----------|-------------------------|---|
| 29 | Streaming job from [output operation 0, batch time 01:26:00] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:26:00 | 0.1 s | 1/1 | 15/15 |
| 28 | Streaming job from [output operation 0, batch time 01:25:55] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:55 | 0.8 s | 1/1 | 18/18 |
| 27 | Streaming job from [output operation 0, batch time 01:25:50] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:50 | 0.4 s | 1/1 | 17/17 |
| 26 | Streaming job from [output operation 0, batch time 01:25:45] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:45 | 0.1 s | 1/1 | 18/18 |
| 25 | Streaming job from [output operation 0, batch time 01:25:40] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:40 | 0.1 s | 1/1 | 17/17 |
| 24 | Streaming job from [output operation 0, batch time 01:25:35] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:35 | 0.1 s | 1/1 | 16/16 |
| 23 | Streaming job from [output operation 0, batch time 01:25:30] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:30 | 0.2 s | 1/1 | 16/16 |
| 22 | Streaming job from [output operation 0, batch time 01:25:25] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:25 | 0.3 s | 1/1 | 12/12 |
| 21 | Streaming job from [output operation 0, batch time 01:25:20] collect at RealTimeTwitterStream.java:58 | 2016/08/15 01:25:20 | 0.2 s | 1/1 | 15/15 |

Figure 4.16: Spark Completed Jobs

4.3.9 Apache Spark Environment Setting



The screenshot shows a web browser window with the URL `localhost:4040/environment/`. The page title is "Environment". It contains two sections: "Runtime Information" and "Spark Properties".

Runtime Information

| Name | Value |
|---------------|---------------------------------------|
| Java Home | C:\Program Files\Java\jdk1.8.0_92\jre |
| Java Version | 1.8.0_92 (Oracle Corporation) |
| Scala Version | version 2.10.4 |

Spark Properties

| Name | Value |
|-------------------------------------|--|
| spark.app.id | local-1471207641641 |
| spark.app.name | Emotive Sentiment Analysis Using spark |
| spark.driver.host | 192.168.1.100 |
| spark.driver.port | 53503 |
| spark.executor.id | driver |
| spark.externalBlockStore.folderName | spark-cdd5f334-da44-419c-863b-033a7aef6265 |
| spark.master | local[4] |
| spark.scheduler.mode | FIFO |

Figure 4.17: Spark Environment Setting

Chapter 5

Conclusion and Future Work

5. Conclusion

This study conveys a remarkable contribution to the literature identifying with the new time series analysis of Twitter content using keyword matching and time series analysis techniques. Firstly, we collect data from twitter through twitter streaming API with twitter OAuth authentication. After that we run spark streaming on twitter stream and run queries on data that are extracted from twitter where query location not null. Get results and put into datasets. Datasets contain ID, Screen Name, Location, Date, Language, Text or Tweet etc. All datasets that contain English language are put into POS Tagger. After POS tagging English Tweets, complete datasets are stored into Mongo DB. Mongo DB document contains field text are stored into text file. Emotion famous words like happy, fear, sadness and joy are given as Synset one by one to WordNet open source tool to extract the synonyms of these emotions and stored in datasets. Text file data match with emotion datasets using naïve Bayes algorithms and count emotions and neutral tweet using word count built into the spark engine class. Emotions positive, negative, neutral and total tweets results are stored into Mongo DB. Extract Mongo DB documents to extract language and location of tweets and the results are stored into Mongo DB. Total three collections that having the results of our data filtration. Emotion collection and language collection and country wise continent collection are stored in JSON format one by one. JSON format results are put into elastic search using a spark. Create an index of every JSON file according to the system date on Kibana. Visualize these Kibana indexes that are extracting data from elastic search. Indexes are display on Kibana dashboards. Also, the previous work targeted only one perspective, historical data or twitter data, but not in real time data analysis human emotions. This study tries to cover both of the views simultaneously with an aim to calculate the outcome of the human emotions worldwide. The whole system

revolves around a collection of data and then evaluating smart features out of it. Huge effort has been driven into the collection and preprocessing of the raw data. This analysis produced some of the energetic features that gave rise to such distinguish results. After that, several machine learning techniques have been applied and the results are analyzed. After performing all of the steps, we analyzed the performance of different models on the same data and it was observed that naïve Bayes theorem worked quite well on the historical data.

5.1 Future Work

In this system, the statistical analysis was carried out to predict the outcome of an emotions, exposes several opportunities for future research work. Also, Machine learning algorithms perform well to classify sentiments in tweets. We believe that the accuracy could even now be progressed. The following is a list of ideas we think could help in this heading.

a. Apache Spark:

Currently, we are working on an apache spark version 1.6.1 and a few days ago, apache launches new version 2.0 that's totally different from version 1.6.1. Performance of apache spark 2.0 is much faster and contains many machine learning algorithms. Spark context is renamed as a spark session in apache spark version 2.0. Streaming visualization in apache spark 2.0 is asynchronous that's not available in spark version 1.6.1. Apache spark version 2.0 is using Scala version 2.11.8 that is much better and mature than Scala 2.10.4 that's available in spark 1.6.1.

b. POS Tagger:

Currently we are working on Stanford NLP POS tagger API for tagging the tweet text. This API understands only English language tweets and analyze the tweet text any other language

except English considered as neutral tweets. It's possible in a future to work on tweets using multiple language POS tagger that understand all the languages and analyze the tweets. So the results will be more accurate and average of neutral tweets will be minimized.

c. WordNet Library uses JAWS:

Currently we are working on project Emotive Sentiment Analysis using the WordNet library version 3.0 that's supports only English language and database available for only English language. This library launch by Princeton university and few years ago stop working on this library. This library provides word details according to parts of speech like happy is adjective and all of these synonyms and antonyms are returned. Moreover, it also returns verb, adverb, noun etc.

d. No SQL Database.

Currently we are working on a project using Mongo DB as a No SQL to store tweets in all the results outcome in a document form. So in a future it's possible to use any other NO SQL database that retrieval time is much faster than Mongo DB and direct support with spark using lambda expression available in Java 8. It has ability to store data in a column format as used in Apache Cassandra NO SQL database.

e. Semantics

We have classified the tweets generally on the basis of the polarity of the words it contains.

The polarity of a tweet may depend upon the perspective you are interpreting the tweet from.

In this case, the semantics may help. Utilizing a semantic role Labeler may demonstrate which noun is mainly associated with the verb and the classification would take place accordingly.

f. Handling neutral tweets

In real world applications, neutral tweets can't be overlooked. Proper POS tagger that's support all languages consider should be minimized to neutral sentiment.

g. Internationalization

Our system concentrates only on English sentences; however, Twitter has numerous international users. It should be possible to use our approach to classify sentiment in other languages for better calculation of the result.

h. Emoticons Smile's (Tone Analyzer)

Currently we are using WordNet tool that provide synset of specific word but not provides the smiles information that are mostly used in a tweet and also too much using in a rating of application in a play store and movies. So in a future emoticons jar or tools if available as open source, then used in a project to mature the outcome of their results and also optimize neutral; tweets that are considered as raw tweet.

Chapter 6

Data Visualization

6. Introduction

Though the overall working of the project mostly focused on the backend processing of the data and the studies doesn't require major time to be spent on the user interface. But for the purpose to visualize the features data and the prediction results, we have used **elastic search** for the indexing of the data and **Kibana** to visualize the data. Different images are included in the chapter referring to the graphical representation of the data and the results achieved by calculating the results of the emotions.

6.1 Methodology

The historical data after filtration was saved into mongo dB this data is used to visualize each emotion performance individually. The data were imported into apache spark through mongo-java driver and using elastic search API for spark we passed the generated RDD from JSON file to the elastic search. After obtaining the data in elastic search, we used Kibana to visualize the obtained data and extracted some of the useful information from it. Some of the information is shown below.

6.2 Graphs

6.2.1 Real Time Streaming in Seconds

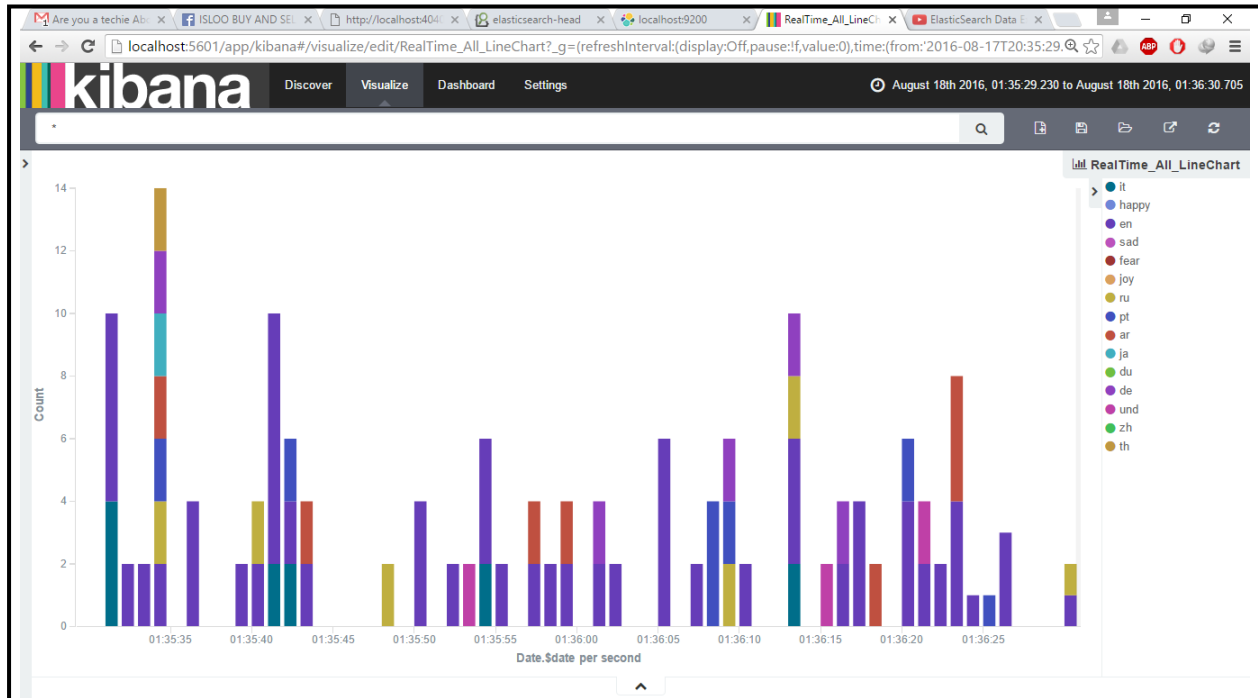


Figure 6.1: Real Time Streaming Per Second for Emotions Happy, Fear, Sad, Joy

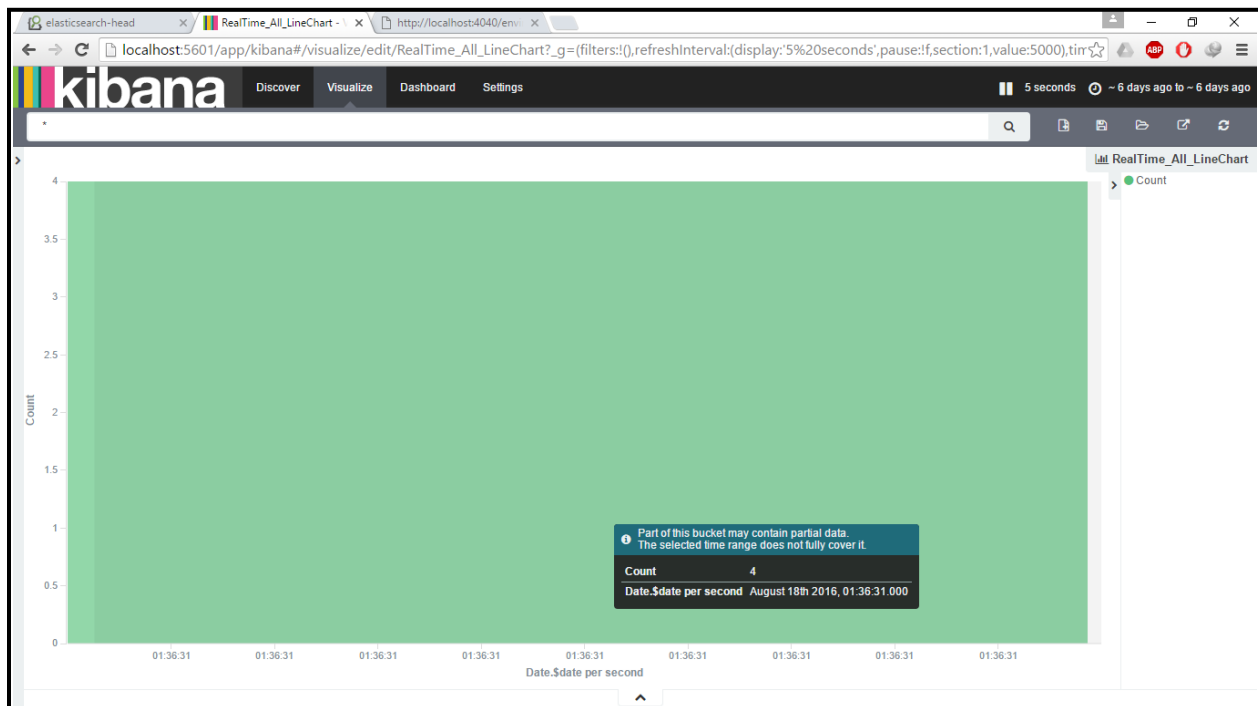


Figure 6.2: Real Time Tweets Four Tweets Per Second Extract

6.2.2 Emotion Results using Different Graphs

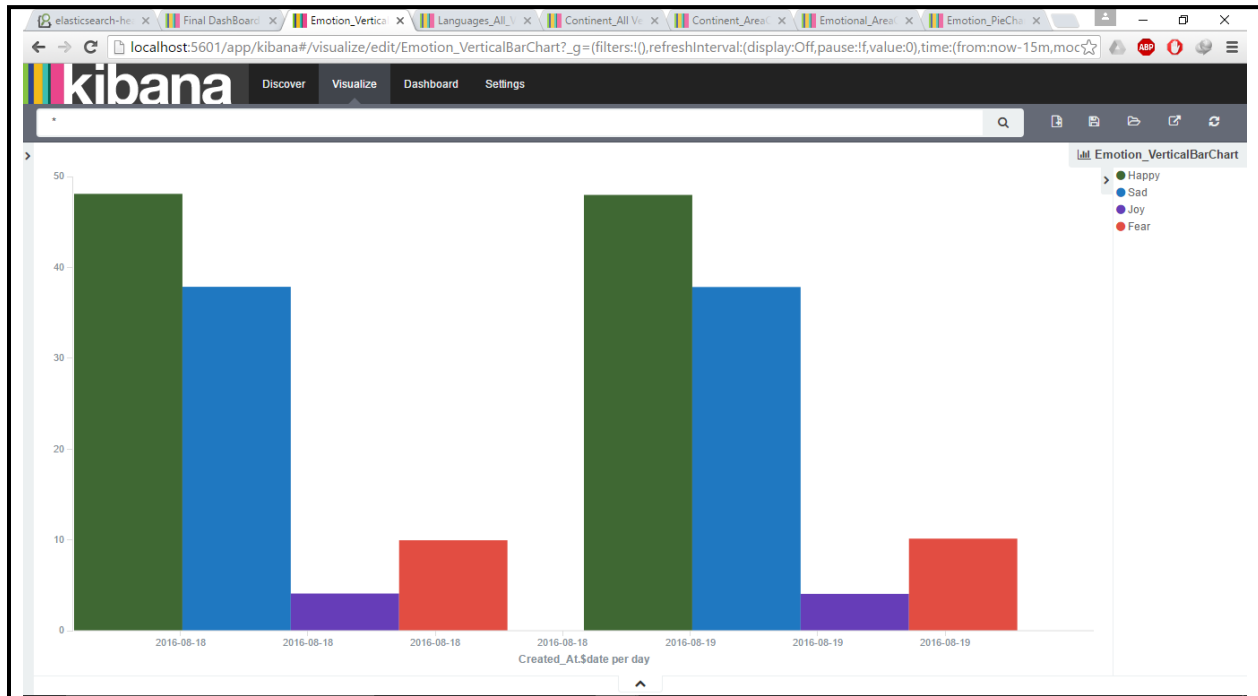


Figure 6.3: Emotions results Per Day Happy, Fear, Sad, Joy

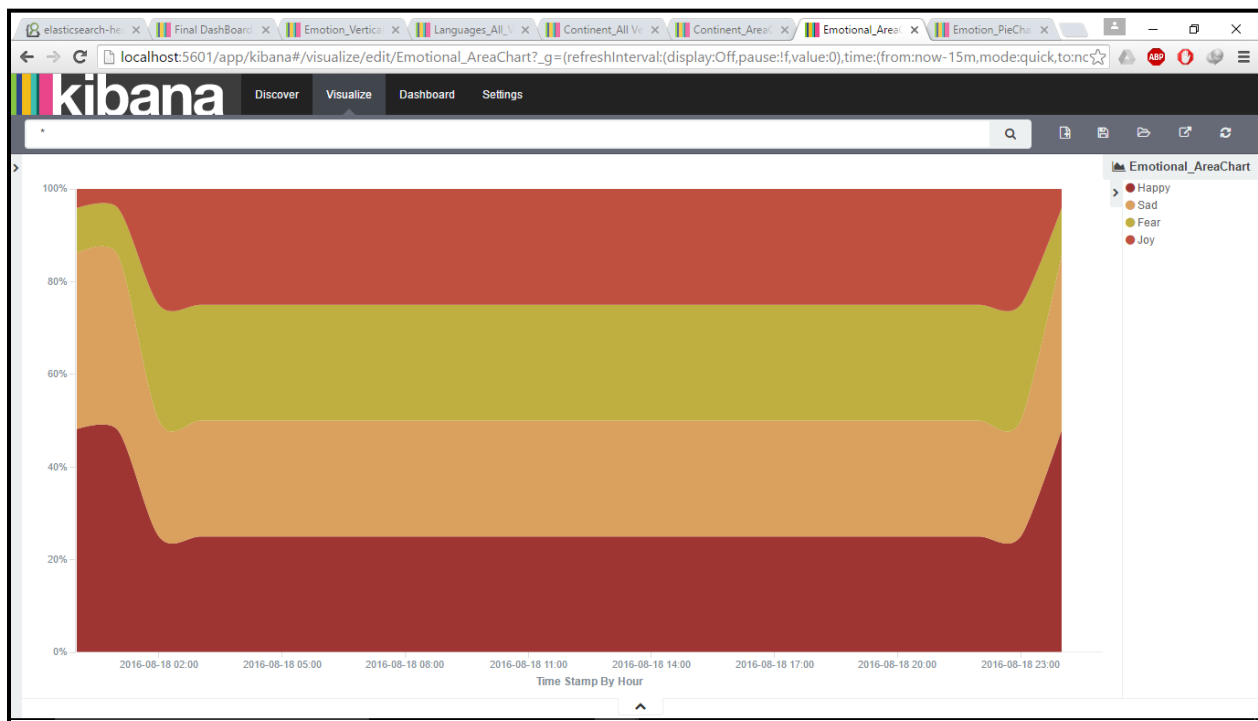


Figure 6.4: Emotion Results Per Hour

6.2.3 Emotion Results using Pie Chart & line Charts

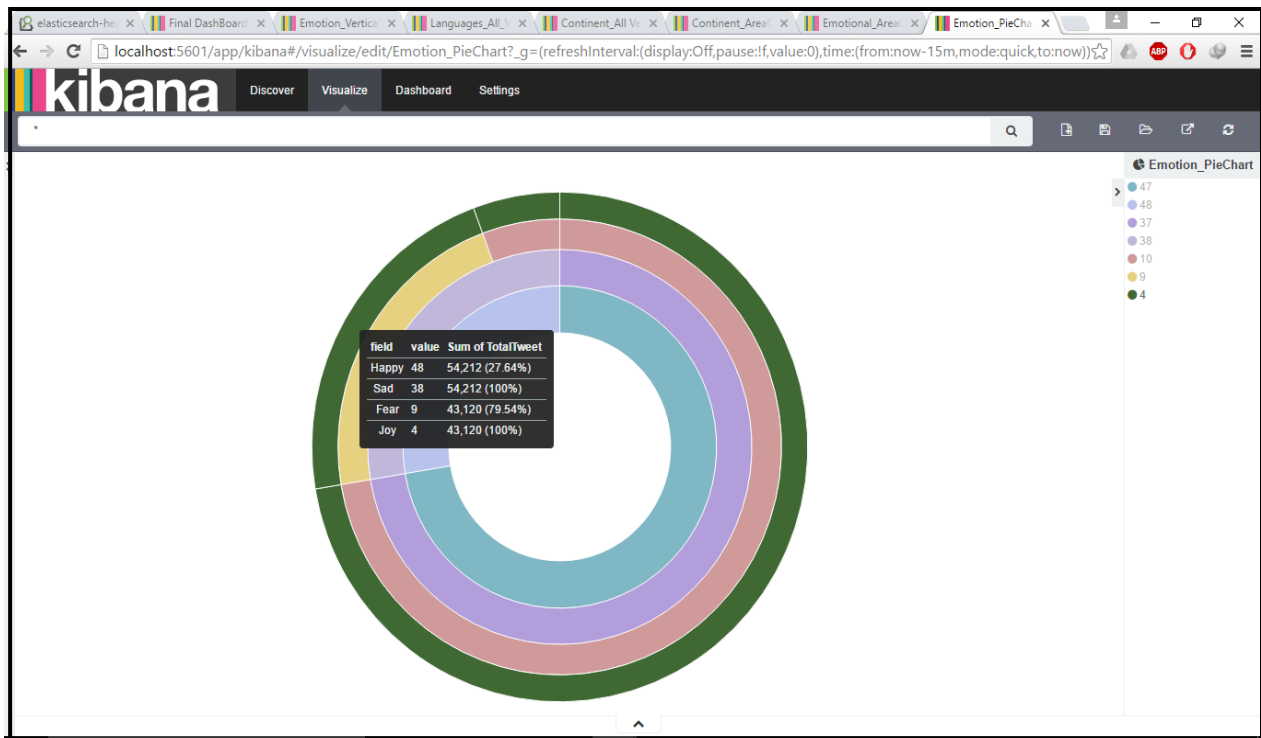


Figure 6.5: Emotions Results Pie Chart Happy, Fear, Sad, Joy by Percentage

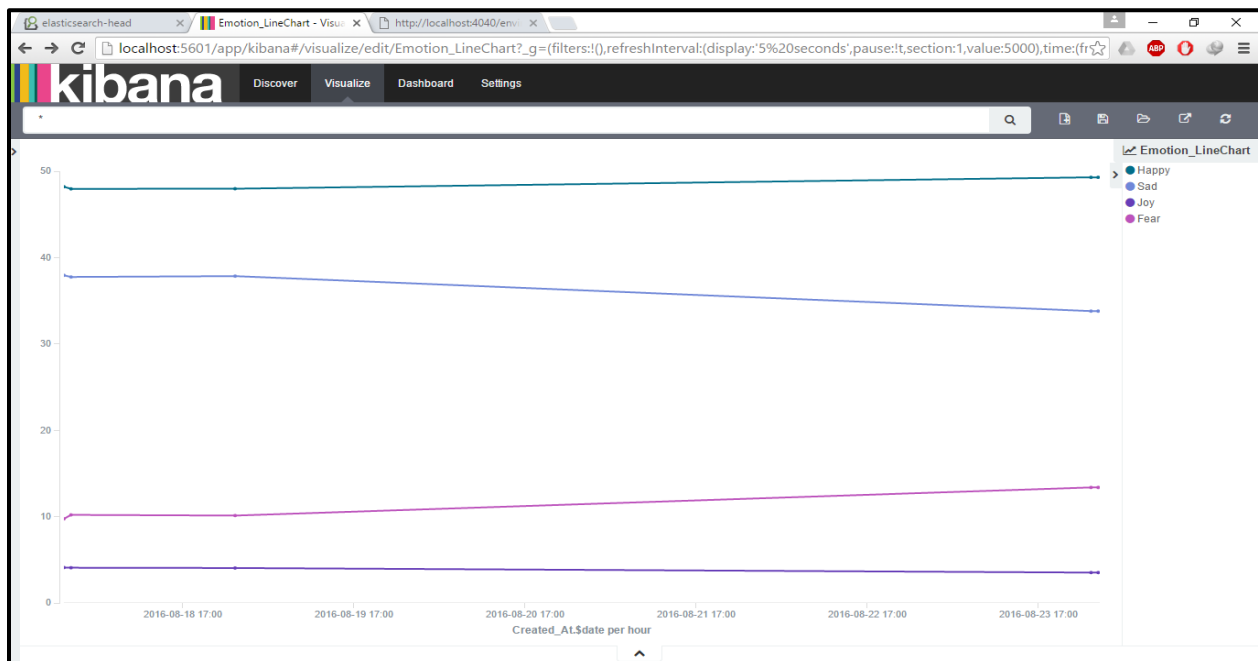


Figure 6.6: Emotion Results Line Chart Happy, Sad etc.

6.2.4 Tweet Graphs by Language

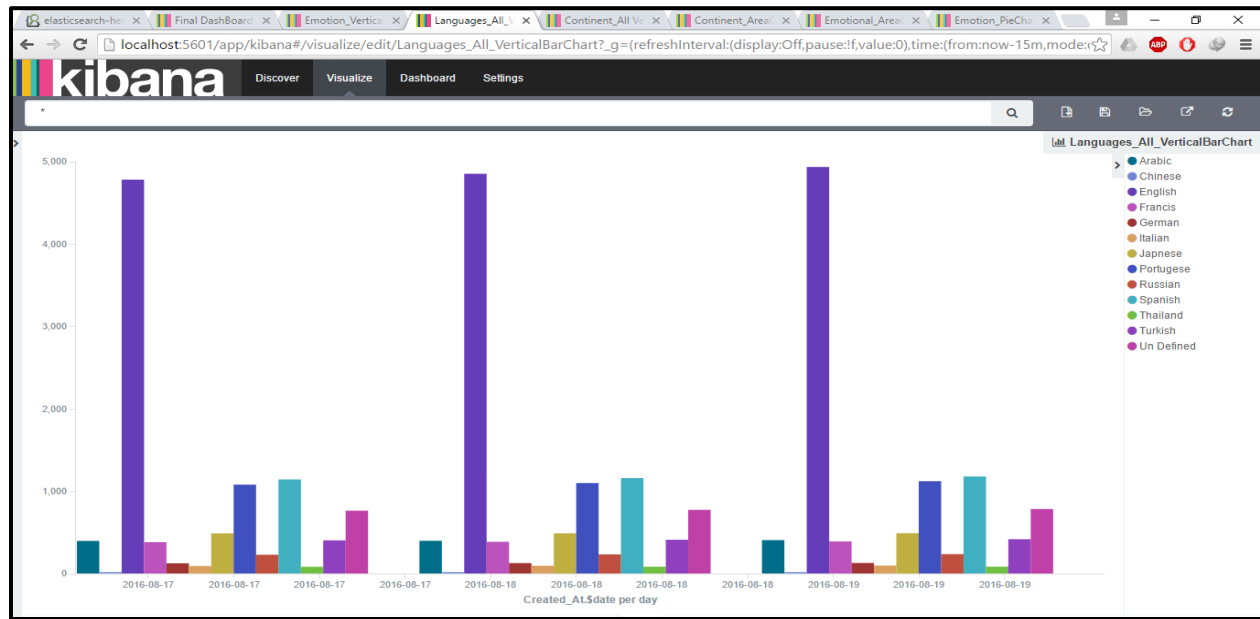


Figure 6.7: Language Results English, Arabic, French, Thailand

6.2.5 Total Tweet Language Result Statuses

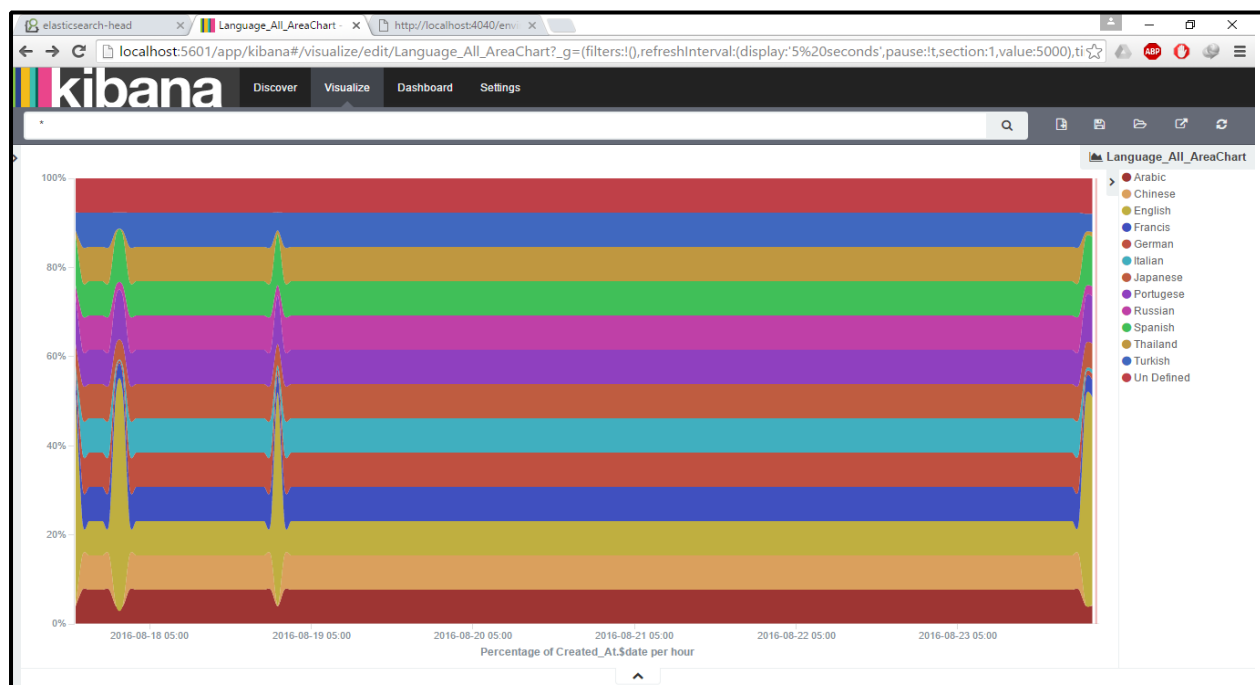


Figure 6.8: Tweet Total Language Results - Red = Arabic, Green = Russia

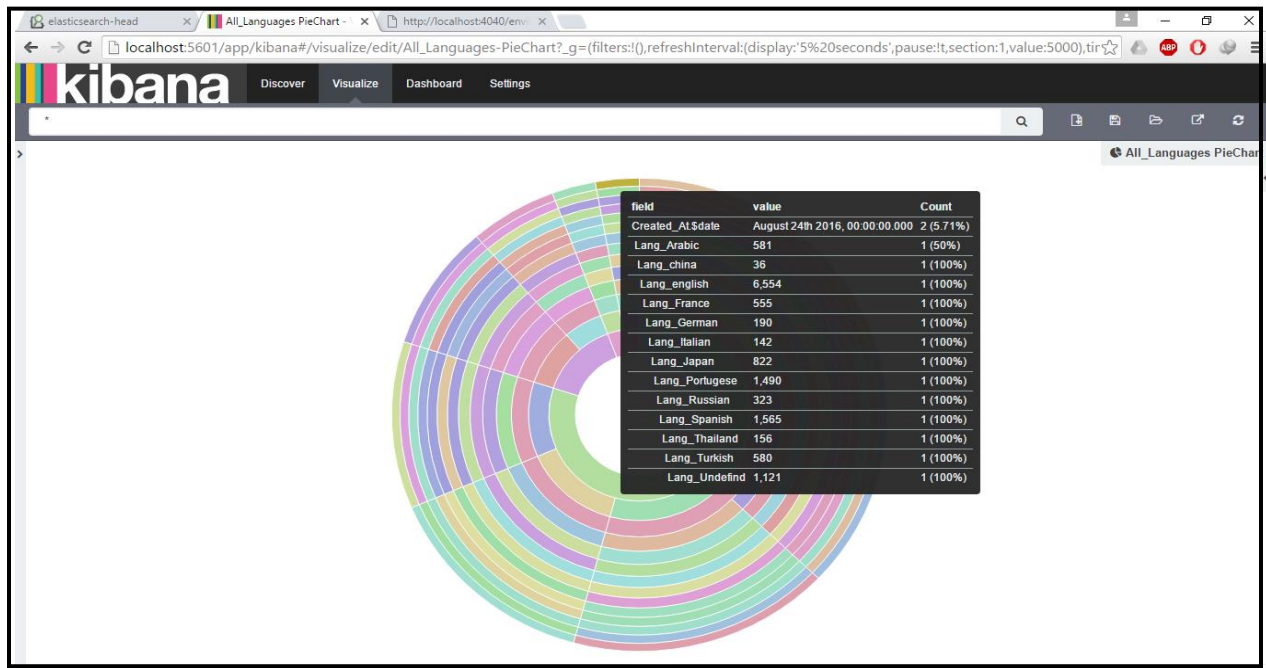


Figure 6.9: Tweet Total Emotions Result – Count Tweet by Tweet

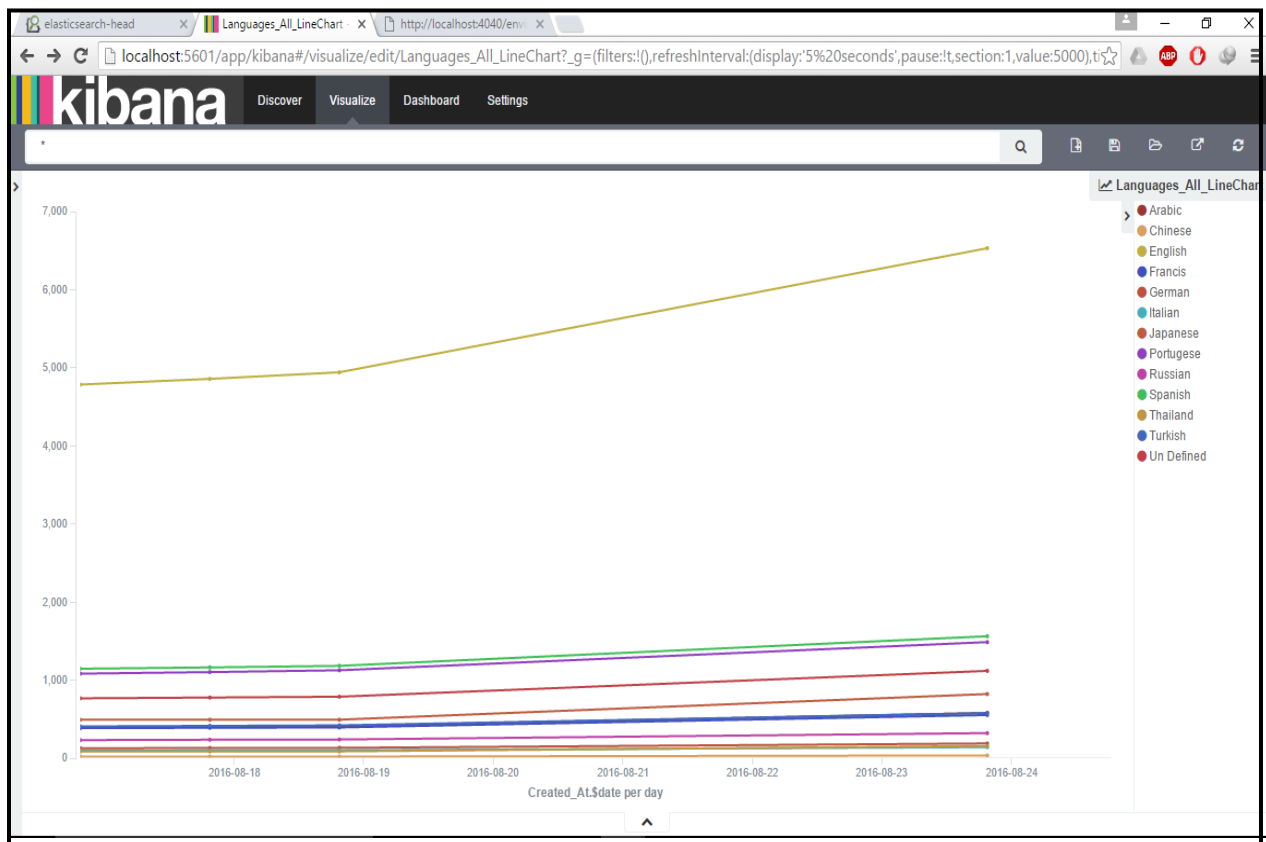


Figure 6.10: Pakistan Tweet Total Result – Language Arabic Italian Spanish

6.2.6 Tweet Graph by Continent

□ Europe-Asia-Oceania-etc.

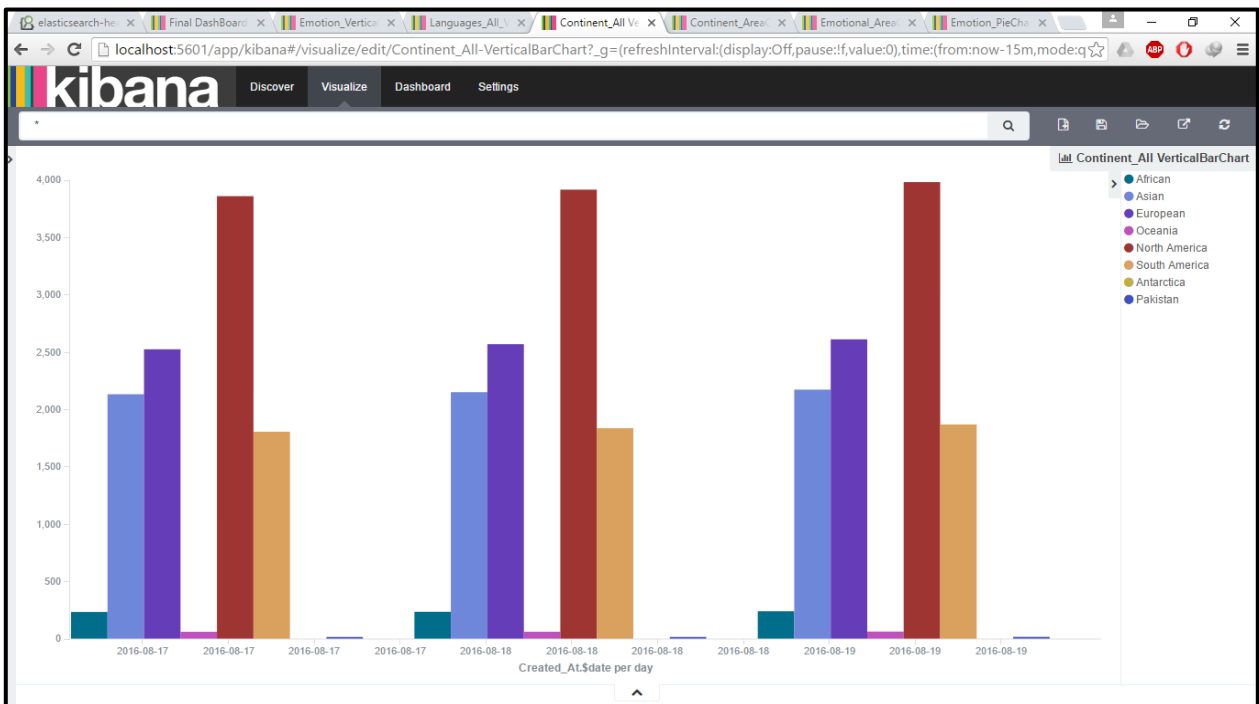


Figure 6.11: Continent of World Africa- Dark green Asia Blue=Europe Navy Blue

□ Sub-Continent Wise Graph

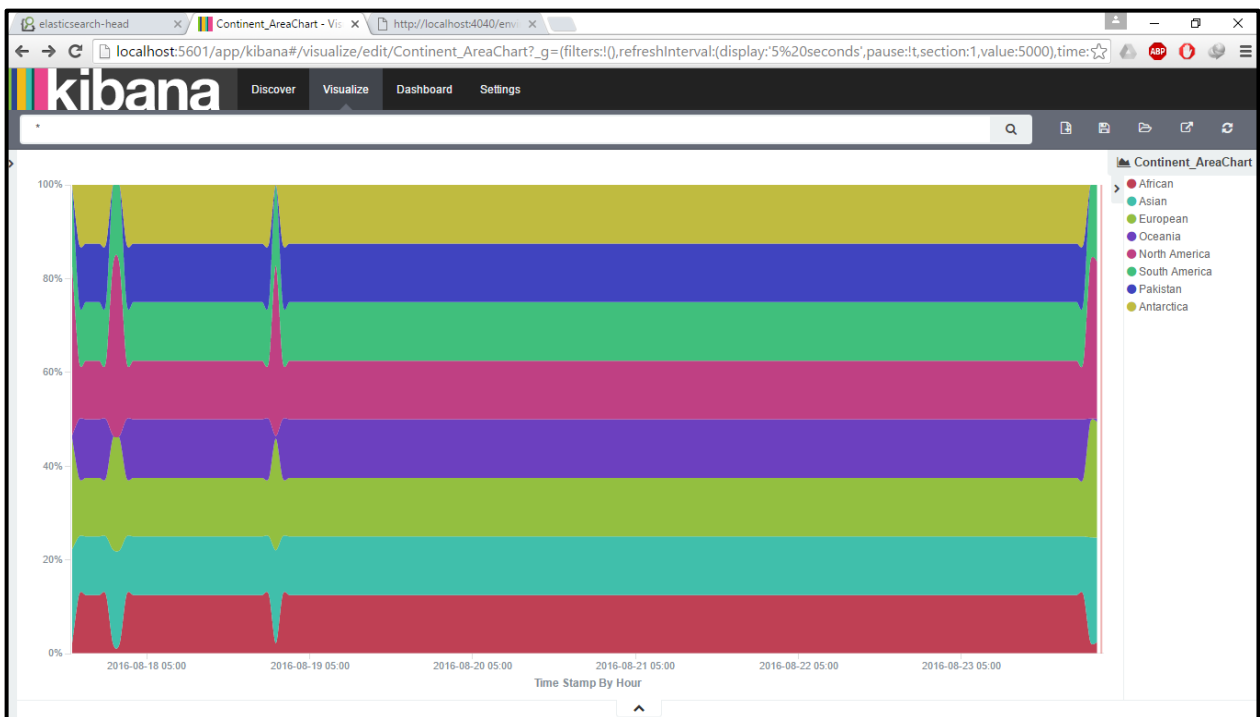


Figure 6.12: Tweet Continent Wise- Yellow=Antarctica Green=Asian Blue=Pakistan Red=African



Continent Pie Graph

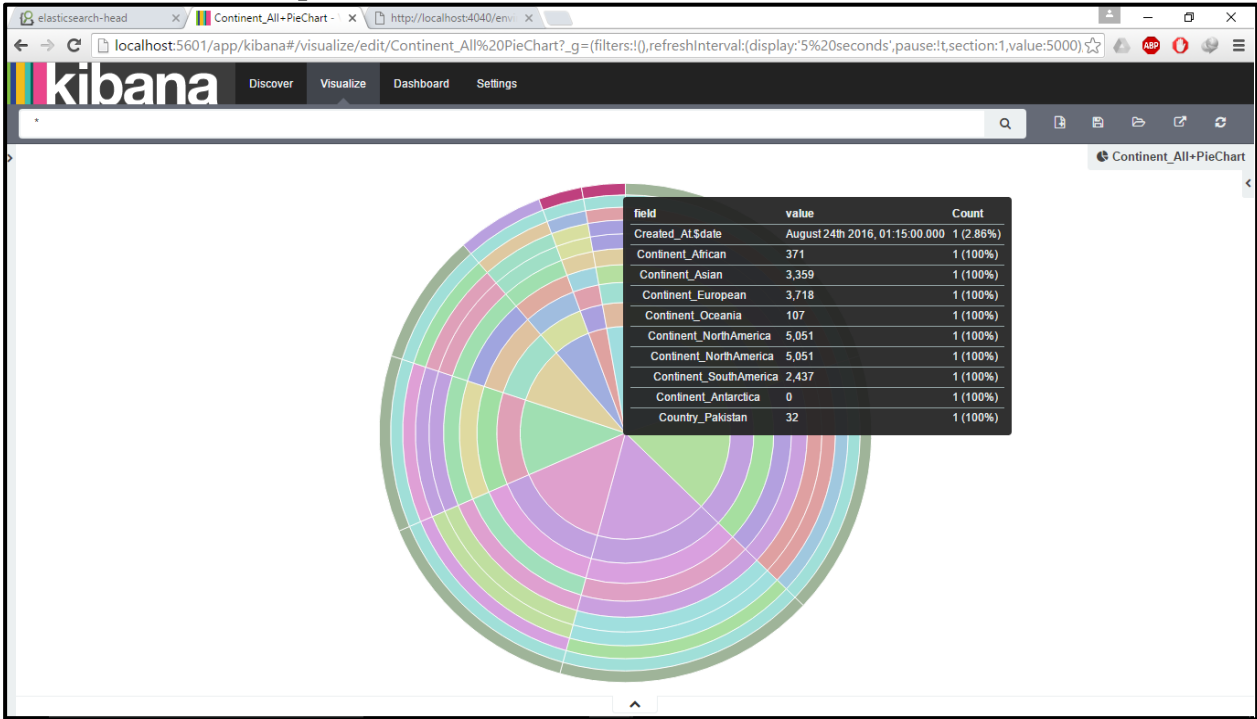
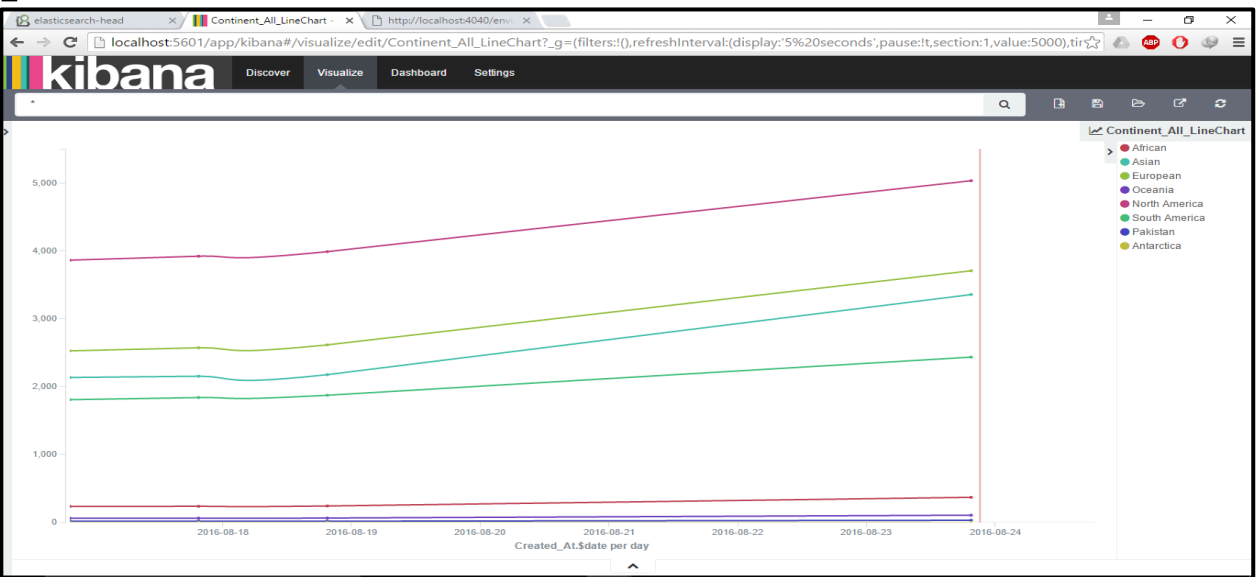


Figure 6.13: Continent Pie Graph Value Count



All World Continent Users



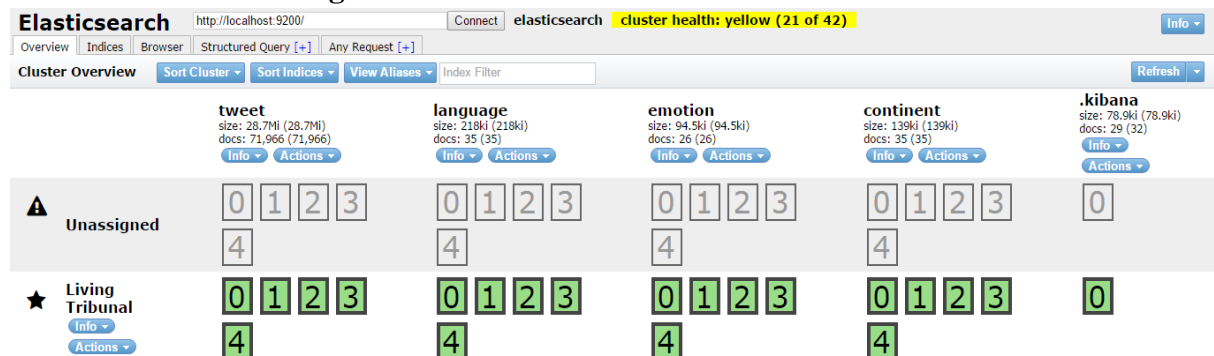
6.2.7 Kibana Dashboard

Final Kibana Dashboard with All Real Time Results



Figure 6.15: Final Kibana Dashboard

6.2.8 Elastic Search Plugin-Head



Chapter 7

Model Testing

7. Introduction

Several methods and techniques were applied on the data with the aim to get the result as accurate we can with the available resources. For the purpose, we applied five different algorithms for the classification of the data. And each algorithm produced a model on the basis of training data respectively. Among these different models, we had to choose a single model that will be used in the future for the actual calculation of the emotion results based on the current situation. The chosen model will definitely be the best among all. We performed numbers of test on the data to determine the performance of the model.

7.1 Methodology

For the purpose to analyze the accuracy of our models, we evaluate the model on the basis of data training and testing ratio, i.e. 70% data for training the model and 30% data for testing. We altered the ratio of five times and analyzed the performance of the model. And then analyzed the model with greater accuracy. The best model was then selected on the basis of its performance on training/testing ratio and was used for predictions. The overall comparison of the models has been shown in the figure below.

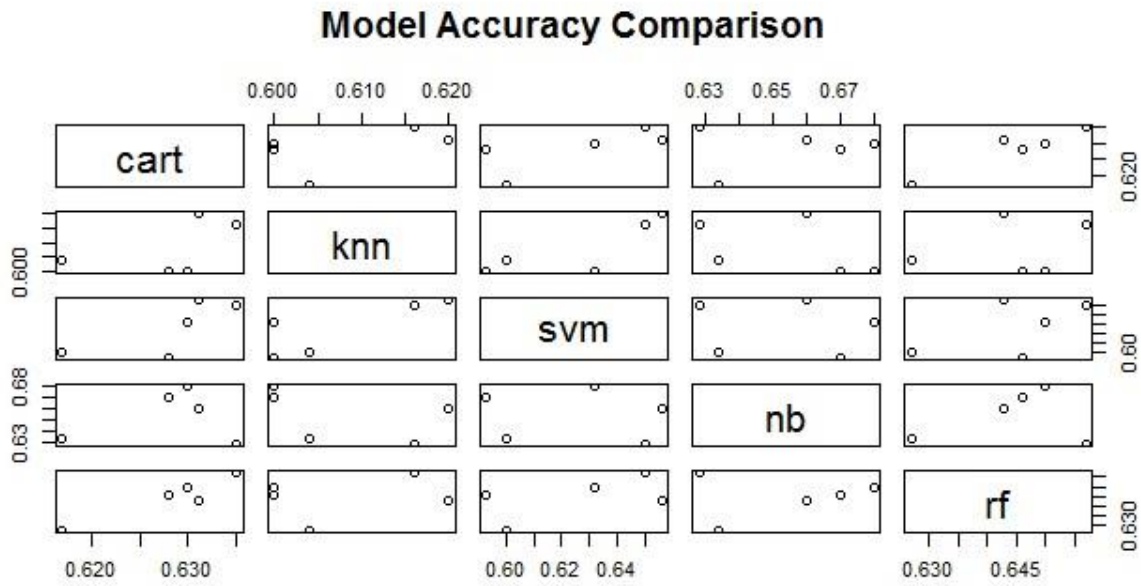


Figure 7.1: Model Accuracy Comparison

7.2 Model Performance Testing (Visualization)

7.3 Naïve Bayes Algorithm (Visualization)

7.3.1 Historical Data

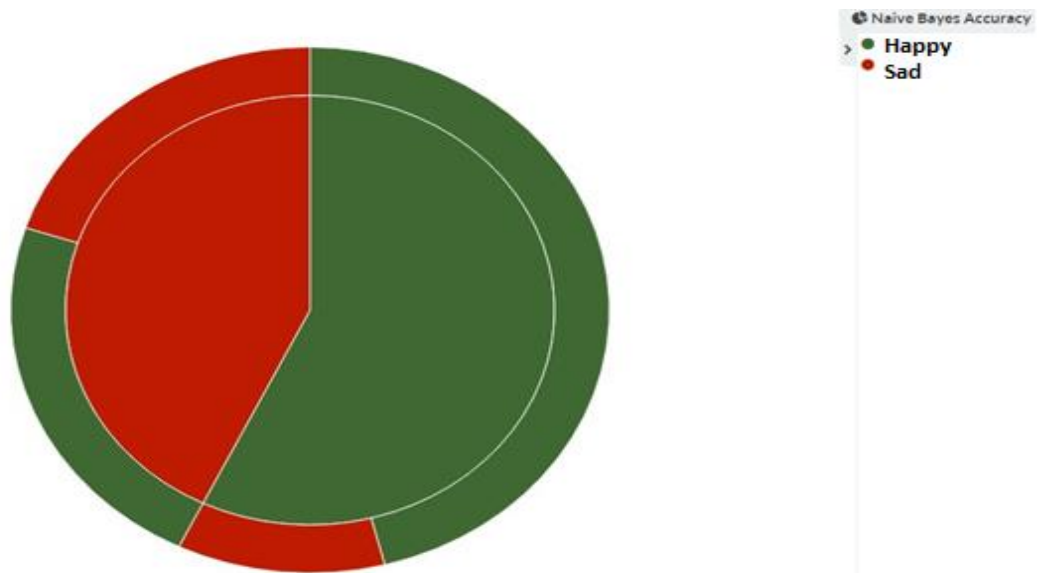


Figure 7.02: Naive Bayes Performance (Historical Data)

7.3.2 Social Media Data

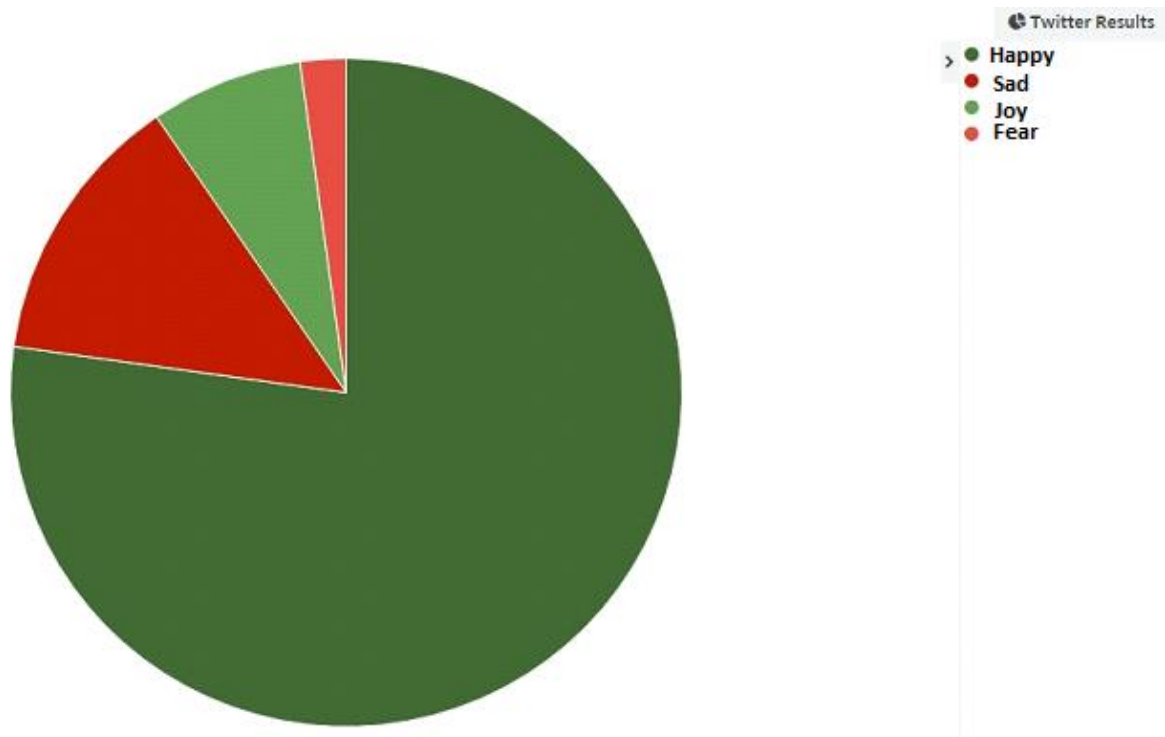


Figure 7.03: Naive Bayes Performance (Social Media Data)

References:

- [1] Effects of the Recession on Public Mood in the UK (Lansdall-Welfare et al., 2012). URL <http://geopatterns.enm.bris.ac.uk/mood/> Date: 20-11-2015:16:40
- [2] Flu Detector tracking epidemics on Twitter URL <http://geopatterns.enm.bris.ac.uk/epidemics/>
Date: 22-11-2015 Time: 15:40
- [3] Earthquake analysis and design vs non earthquake analysis and design using staad pro:
E-ISSN 0976-3945
- [4] Stanford Log-linear Part-Of-Speech tagger URL <http://nlp.stanford.edu/software/tagger.shtml>
Date: 28-04-2016 Time: 16:40
- [5] WordNet: An Electronic Lexical Database URL <http://mitpress.mit.edu/books/wordnet>
Date: 29-04-2016 Time: 19:30
- [6] Apache Spark Documentation URL <http://spark.apache.org/documentation.html>
Date: 19-05-2016 Time: 18:40
- [7] Submitting Applications on Standalone Cluster URL: Date: 29-04-2016 Time: 16:40
<https://spark.apache.org/docs/1.6.1/submitting-applications.html>
- [8] JAWS Library javax.jws-1.0 <http://www.java2s.com/Code/Jar/j/Downloadjavaxjws10jar.htm>
Date: 15-08-2016 Time: 12:40
- [9] J. Bollen, H. Mao and X. Zeng. Twitter mood predicts the stock market. Journal of computational science, 2011.
- [10] S. Golder and M. Macy. Diurnal and seasonal mood vary with work, sleep, and day length across diverse cultures. Science, 333(6051):1878–1881, 2011.