TRACKING MENTAL HEALTH USING SOCIAL MEDIA

by

MANISH RANJAN

(Under the guidance of Shannon Quinn)

Recently, “Big Data” techniques have been successfully used to solve challenging problems in healthcare. Such techniques have given rise to the development of “biosurveillance” frameworks. These frameworks are an application of big data processing paradigms which addresses the problem of identifying and predicting threats to public health. However, existing biosurveillance platforms are limited in their applicability to task such as detecting seasonal outbreaks of flu or specific mental disorder conditions like schizophrenia. We present a biosurveillance framework that not only anticipates public health threats, but can identify at risk individuals from social media for non contagious diseases in the realm of mental health disorders. In our proposed framework, we combine topic modelling with sentiment analysis to provide an estimate of toxic or abusive behavior, identifying a pool of potentially at-risk users via their content on social media.  This framework can be tuned on incoming data incrementally over a period of time, which ensures better result over time on unobserved data. In a more mature phase our framework could be used by medical professionals to monitor and study users for their mental health illness/disorders more closely and accurately.

INDEX WORDS:     Mental Health, Behavioral health, health-surveillance, Twitter, Data Mining, Scalable Machine Learning, wellness, Topic modelling, NLP techniques.

TRACKING MENTAL HEALTH USING SOCIAL MEDIA

by

MANISH RANJAN

B.TECH., SASTRA UNIVERSITY, INDIA, 2008

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2016

© 2016

Manish Ranjan

All Rights ReservedTRACKING MENTAL HEALTH USING SOCIAL MEDIA

by

MANISH RANJAN

Major Professor: Shannon Quinn

Committee: Lakshmish Ramaswamy

Timothy G. Heckman

Electronic Version Approved:

Suzanne Barbour

Dean of the Graduate School

The University of Georgia

December 2015

DEDICATION

This is for you, xxxx.

ACKNOWLEDGEMENTS

I would like to thank Dr. Shannon Quinn for his guidance, not only in this

project, but also in my overall graduate academic career. I would also like to thank Dr.

Lakshmish Ramaswamy from whom I learned a lot during my two fruitful years at

University of Georgia. I would like to thank Dr. Timothy Heckman for accepting to

provide me guidance. I would also like to thank Al Aila, BahaaEddin, a MS in Artificial Intelligence student at the University of Georgia for all those white board discussions. Lastly, I would like to thank all my professors from whom I have learnt a lot throughout my academic career.

TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS v

LIST OF TABLES viii

LIST OF FIGURES ix

CHAPTER

1 Introduction 1

1.1 Introduction 1

1.2 Challenges 4

1.3 Major Contribution 4

2 Background 7

2.1 Twitter and its characteristics 7

2.2 Use of Machine Learning in Twitter 9

2.3 CyanoHABs and Meghdoot/Cyanotracker Project 10

3 Related Work 14

3.1 Disaster Management using social media 14

3.2 Disease Monitoring using social media 21

4 System Architecture and Overview 23

4.1 Challenges 23

4.2 System Architecture 25

4.3 Data Collection 26

5 Keyword Analysis 28

6 Machine Learning Analysis 38

7 Conclusion 54

REFERENCES 56

LIST OF TABLES

Page

Table 1: Number of Tweets 26

Table 2: Rate and Percentage of Relevant and Irrelevant Tweets 30

Table 3: Unique, Reliable and Active Users for each hashtag 31

Table 4: Correctly Classified Instance for two algorithms and average for each hashtag 45

Table 5: F-Measure for two algorithms and average for each hashtag 45

Table 6: Recall for two algorithms and average for each hashtag 46

LIST OF FIGURES

Page

Figure 1: CyanoTracker Architecture 12

Figure 2: System Architecture and Overview 25

Figure 3: Number of Relevant and Irrelevant Tweets 31

Figure 4: Top-20 frequent words in the Description of Relevant Users (Red Tide) 34

Figure 5: Top-20 frequent words in the Description of Relevant Users (Cyanobacteria) 34

Figure 6: Top-20 frequent words in the Description of Relevant Users (Blue Green Algae) 35

Figure 7: Top-20 frequent words in the Description of Relevant Users (Algae Bloom)...35

Figure 8: Same month Experimental Setup for Machine Learning Characteristics …….40

Figure 9: Different month Experimental Setup for Machine Learning Characteristics …40

Figure 10: Average F-Measure Vs. Difference in Months ……………………………...47

Figure 11: Average Correctly Classified Instances Vs. Difference in Months ………….47

Figure 12: Average Recall Vs. Difference in Months …………………………………...48

Figure 13: Jaccard’s Coefficient Vs. Difference in Months …………………………….49

Figure 14: Precision improvement after using retraining model (Cyanobacteria) ……...51

Figure 15: Precision improvement after using retraining model (Algae Bloom) ……….51

Figure 16: Precision improvement after using retraining model (Red Tide)……………52

CHAPTER 1

INTRODUCTION

**Come back to this** “In this chapter, we have briefly talked about the growing popularity of online social media, and also discussed how microblogs like Twitter, are used as data gathering platforms in citizen science activities. We have also described major challenges associated with social media for data analysis and our contributions for the same.

**1.1 Introduction**

Being mentally healthy is defined by World Health Organization (WHO) as state of well being in which (1) every individual realizes her or his own potential, (2) can cope with the normal stresses of life, (3) can work productively and fruitfully, and (4) is able to make a contribution to her or his community. However, having mental illness is a rather more serious problem. National Institute of Mental Illness (NAMI’s) report suggest that there are 43.8 million adults experiencing mental illness per year. NAMI’s latest reports also indicate that,

* 1 in every 5 adults in America experiences a mental illness.
* Nearly one in 25 adults in America live with a serious mental illness.
* One half of all the chronic illnesses begin by the age of 14 and three quarter by age of 30.

Above data suggests mental disorders are common in the United States. However, mental

disorder can be categorized as a disease only if a person experiences disability due to

serious mental illness (SMI). The criterion to define SMI is as follows as per National

Survey on Drug Use and Health (NSDUH).

* A mental, behavioral, or emotional disorder (excluding developmental and substance use disorders)
* Diagnosable currently or within the past year;
* Of sufficient duration to meet diagnostic criteria specified within the 4th edition of the Diagnostic and Statistical Manual of Mental Disorders(DSM-IV); and,
* Resulting in serious functional impairment, which substantially interferes with or limits one or more major life activities.

Given the widespread presence of variations of mental disorder, tracking it from its early onset with a generic approach is very important.

**1.2 Why tracking through social media makes sense**

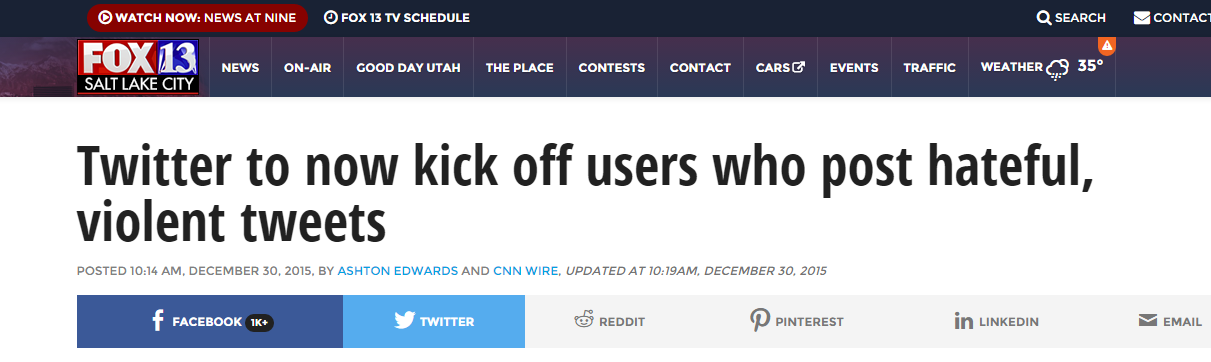
NAMI’s statistics suggests that mental illness is a problem which affects younger demographics more than adult ones. However, mental illness does not manifest the same way as more traditional disease like influenza; often there are no physical symptoms until it is too late.

The spectrum of disorder consists of: anxiety, mood, psychotic, eating, addiction, personality, obsessive compulsive disorders (OCD), Post-traumatic Stress disorders (PTSD) etc. [1]. Here, anxiety disorder is chronic mental condition characterized by an excessive and persistent sense of apprehension. Mood disorder is another psychological disorder characterized by the elevation or lowering of a person's mood, such as depression or bipolar disorder, to define a few.

Many states have already started the process to set up “Bio Surveillance” systems to detect epidemics, finding new health topics and trends using social media as input to harvest upon the social data derived knowledge power. US Department of Health & Human Services (<http://nowtrending.hhs.gov/>) is one such example.

However, being able to set up a broader criterion like “negativity over all” or “sharing toxic content” is not yet solved in a meaningful way. There has been some work like “Tracking Mental Disorders Across Twitter Users” [2]. It is a quality work but it restricts itself to very specific mental health issues like depression. Further, approach is naive, non-scalable and relies on crowdsourcing to verify accuracy. A potential problem with crowdsourcing with respect to tagging mental illness is, it is a very subtle problem in early phases. Detection by just looking at text is difficult even for very experienced psychiatrist. Crowdsourcing such detection hence is definitely not an accurate way to go forward. “Economic Costs of Alcohol and Drug Abuse and Mental Illness” 1985 is another sound theoretical work from an economist’s perspective, but suffers from the lack of readily available implementation strategy.

**1.3 Problem recognized for its seriousness by ‘Tech Giants’**

****

On the same line, Apple has released an app named “HealthKit” for tracking mental

health of individuals in last quarter and has shared plans on roadmap of this app as well.

**1.4 Challenges in Social Media**

Tracking mental health behavior in an online age could be attempted by tracking social media posts. Online social data has in general has two famous outlets, blog posts and Twitter/Facebook. Blog posts are   very difficult to sample content from a large number of unique users. One possible way to use blog posts is to connect with 4k sampled unique user’s blog individually and download their content. This is a non scalable approach since it involves detection of user of interest, finding their blog posts and then downloading content one by one.

Twitter on the other hand makes it very easy with its open interface architecture. We can plugin the API to get access to samples from a public stream.

Despite twitter’s ease of access to data, it has few unique challenges. Firstly, Twitter provides a very conducive platform for the rapid development of online rhetoric. These rhetorics neither appear in dictionary nor are formally recognized as a part of language. But it carries semantic meaning to large number of users and it is therefore critical to identify user’s intent from the rhetorics. Secondly, 140 characters’ limitation on tweets forces users to place many words together with incomplete and incorrect spellings. This makes data processing and building models in NLP domain a challenge. This is partially unsupervised problem in some sense, as we do not have a priori knowledge of all possible terms that are indicative of mental well-being. Furthermore, these terms are changing constantly, as new terms appear on the fly in response to changing circumstances.

**1.5 Major Contribution**

We wanted to study users who were filtered under broader criterion like “negativity over

all” or “sharing toxic content”. Further we wanted to study in detail if sharing of negative

and toxic content on social media could be a potential proxy for underlying mental

illness.

We focus on such filtered users based on not only toxic or negative content they

share, whom they follow, who follows them and what signals they choose to amplify.

Along with our own sentiment analysis algorithm, by adding network features, we seek to improve detection accuracy.

For detection of such user at high scale, we use generative statistical model Latent Dirichlet Allocation (LDA) from Natural Language Processing (NLP) domain, which allows set of observations to be explained by unobserved groups. Our goal is to design a scalable pipeline which could, use twitter data in holistic way, using scalable machine learning technique. Our pipeline should also tune its learning by training on incoming data incrementally over a period of time. Incremental learning would ensure better results with time on unseen test data.

We wanted to avoid using crowdsourcing as validation mechanism. Mental illness in early stages is really subtle and has little or no physical symptoms. Hence, even experienced psychiatrists find it difficult to diagnose it accurately. Hence we could not have left it to internet to decide how good the model was. Another issue with crowdsourcing is, it’s not a scalable approach. We consciously wanted to pick only scalable components for this pipeline, because we wanted the pipeline to work at scale of Twitter [13].

**1.6 Broader Impact**

One such pipeline could not only filter negative users at a very early stage but also unearth hidden pattern associated with negative content on social media.

One such pipeline idea is noble which uses NLP’s generative statistical model technique to filter and create unsupervised cluster of users, is capable of classify unseen users based on content of tweets, could self learn with time to improve accuracy and fall backs on an improved sentiment analysis algorithm for validation. [one potential work could be including feedback]

The rest of the thesis is organized as follows: CHAPTER 2 describes characteristics of Twitter, use of machine learning algorithms on Twitter data and also describes the architecture overview of CyanoTracker project. CHAPTER 3 describes related work, CHAPTER 4 gives an overview on system architecture followed by Keyword Analysis and Machine Learning Analysis in CHAPTER 5 and CHAPTER 6. CHAPTER 7 gives the conclusion of the analysis we have performed. [To be changed]

CHAPTER 2

BACKGROUND

In this chapter, we have briefly described different characteristics of Twitter, which play an important role in social media analysis. Also, we have described the importance of machine learning algorithms on Twitter Data. We have also given an overview of the CyanoTracker project initiated by the researchers at the University of Georgia. [**to be changed**]

**2.1 Social Media and its characteristics**

Tracking mental health behavior in an online age could be attempted by tracking social media posts. Online social data has in general has two famous outlets, blog posts and Twitter/Facebook. Blog posts are   very difficult to sample content from a large number of unique users. One possible way to use blog posts is to connect with 4k sampled unique user’s blog individually and download their content. This is a non scalable approach since it involves detection of user of interest, finding their blog posts and then downloading content one by one.

Twitter on the other hand makes it very easy with its open interface architecture. We can

plugin the API to get access to samples from a public stream.

Last 4-5 years we have seen a huge surge in effort to design good distributed systems

[11], [14] which can handle data at large scale. These systems are also designed to

consume huge volume of data to train machine learning algorithm which are specifically

designed to work in distributed setup. This has created opportunity for researchers and

scientist to create bio-surveillance systems which are predictive in nature and consume

social media as input. Some noticeable works are predicting PTSD [10], Postpartum

changes [15] using social media as input to name a few. While there has been lot of

quality work using twitter as a medium to detect and predict specific mental health

disorders, we will use this section to mention few of the most correlated with our problem

definition.

As a part of literature survey, we investigated different research papers related to mental disorders and use of social media like Twitter and Facebook as user put an increasing amount of personal information on these platforms. Research at Johns Hopkins to use Twitter to track the flu [1] and tweets analysis to provide insight in to metal illness [2] have established methods that can link the content of tweets to disease outbreak as well as specific mental disorders. Another notable work is around discovering co-occurrence Patterns of Asthma and Influenza [3]. There has been lot of work on finding accurate sentence sentiment of given short text using multiple techniques. However, work by Sara Rosenthaon combined study of many popular algorithms to find most accurate sentiment [4] was most comprehensive. The impact of celebrities’ influence on their followers was also studied in detail [5]. Looking at related area research papers, we have concluded to the best of our knowledge, that no direct work has been done to detect toxic content sharing among people. Further, most of the early attempts are very specific to one mental conditions. [8][9] [10]

**2.2 Sentiment Analysis**

Sentiment analysis has been handled as NLP task at many levels. Starting from document level classification task to (Turney, 2002; Pang and Lee, 2004), to at the sentence level (Hu and Liu, 2004; Kim and Hovy, 2004) and most recently at the phrase level Wilson et al., 2005; Agarwal et al., 2009).

However, for a more detailed and summarized study of role of social media in mental health research, we would like to refer users to De Choudhury 2013. De Choudhury has identified many ways in which NLP can be used to identify and predict mental health issues both at individual and population level.

For population-level analysis, surveys such as the Behavioral Risk Factor Surveillance

System (BRFSS) are conducted via telephone (Centers for Disease Control and

Prevention (CDC), 2010). Some of these surveys cover relatively few participants (often

in the thousands), have significant cost, and have long delays between data collection and

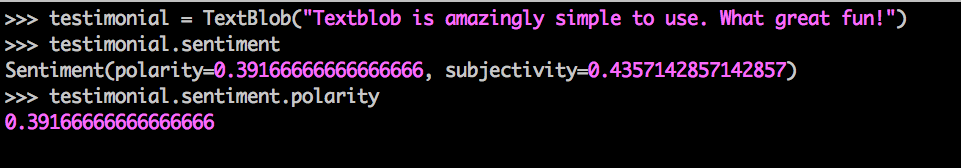
dissemination of the findings.

We were interested in usage of negative and positive words used by users. at the same time, we also wanted to know the distribution of positive and negative score for each individual. This was to get an overall sense of mental health by combining it with user’s network feature like who follows user, user follows whom, what is the impact of celebrity count.

We zeroed down on two approaches to handle sentiment analysis. (1) To use TextBlob package. [26] (2) We wrote a python piece of code which works on bag of word model to assign sentiment to a word based on a published sentiment scores file [27]. We will get in to detail of our implementation later in chapter [x] section [y]. We wanted two different approaches one taking sentence context in to consideration and another just using bag of word model.

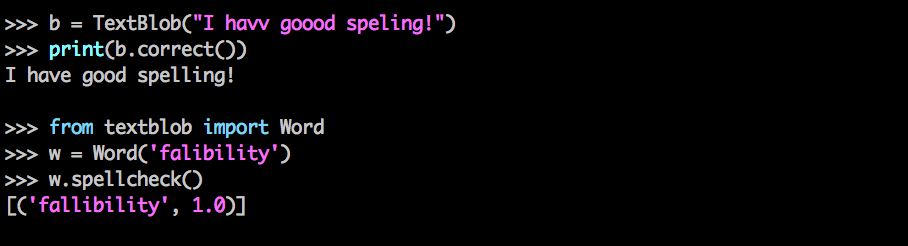
TextBlob’s [26] python package’s sentiment property returns a namedtuple of the form Sentiment (polarity, subjectivity). The polarity score is a float within the range [-1.0, 1.0]. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective. It could take a sentence and return the sentence score.

e.g.



Reason behind using TextBlob’s sentiment package was it had really consistent and rich set of Application Programming Interfaces (APIs). Additionally, this decision was also influenced by TextBlob’s API’s capability to correct word spelling and spellcheck.

e.g.



However, while the accuracy of TextBlob was really good, speed was very slow. We believe that it also had something to do with data quality we had. For instance, if we had to look in to a formatted text doc and correct the spelling, the no of occurrences API will have to invoke corrector method will be very less. However, in case of tweets, because its very common to not pronounce word correctly, it was just proving to be overhead. Hence we moved on with our own Dictionary based scorer, which will be discussing in detail in chapter [x] section [y].

**2.3 Latent Dirichlet Allocation (LDA)**

In this section we will briefly topic modelling approach LDA[28] we used. In later section [x] we will introduce the adaptation of this model for twitter data.

LDA is an unsupervised machine learning technique whose primary purpose is to identify latent topics and word probability distribution for those topics from a large document collection. In LDA each document is represented as probability distribution of topics while each topic itself is probability distribution of words. LDA as a model allows us to play around with number of topic we expect the document to produce and also word in each topic given a word corpus as input. This also clearly indicates that LDA is designed on bag of word concept. In context of this problem, each document here is created by preprocessing user’s last accessible tweets. This kind of unsupervised topic modelling was especially suitable for us as we were looking at generic negative content and later how well each user’s tweet (document) was correlated with topics LDA has figured out.

**2.4 Unsupervised Clustering Techniques**

Two unsupervised clustering used in this experiments to cluster users are:

1. K-Mean [30]
2. Expectation Maximization (EM) [31]

The basic rationale behind selection these two algorithms was to find if the users have clear boundary when they are clustered together (K–Mean would work well) or they have fuzzier distribution. (EM should work well). We used “sklearn. cluster” [32] module of python to perform these clustering operation.

K means

1. Hard assign a data point to one particular cluster on convergence.
2. It makes use of the L2 norm when optimizing (Min {Theta} L2 norm point and its

centroid coordinates).

In contrast, EM

1. Soft assigns a point to clusters (so it gives a probability of any point belonging to any centroid).
2. It doesn't depend on the L2 norm, but is based on the Expectation, i.e., the

probability of the point belonging to a particular cluster. This makes K-means

biased towards spherical clusters.

**2.5 t-SNE (t-Distributed Stochastic Neighbor Embedding)**

t-SNE is a machine learning algorithm for dimensionality reduction developed by Laurens van der Maaten and Geoffrey Hinton. It is a nonlinear dimensionality reduction technique that is particularly well suited for embedding high-dimensional data into a space of two or three dimensions, which can then be visualized in a scatter plot. Specifically, it models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points [33].

It was well suited for visualization of our user’s cluster as they were expected to have high dimensional data as each topic is one of the dimension if user’s feature vector. We will get in to finer details of our user feature vector in section [x].

CHAPTER 3

**PIPELINE ARCHITECTURE**

Here is an Architectural overview at a very high level.

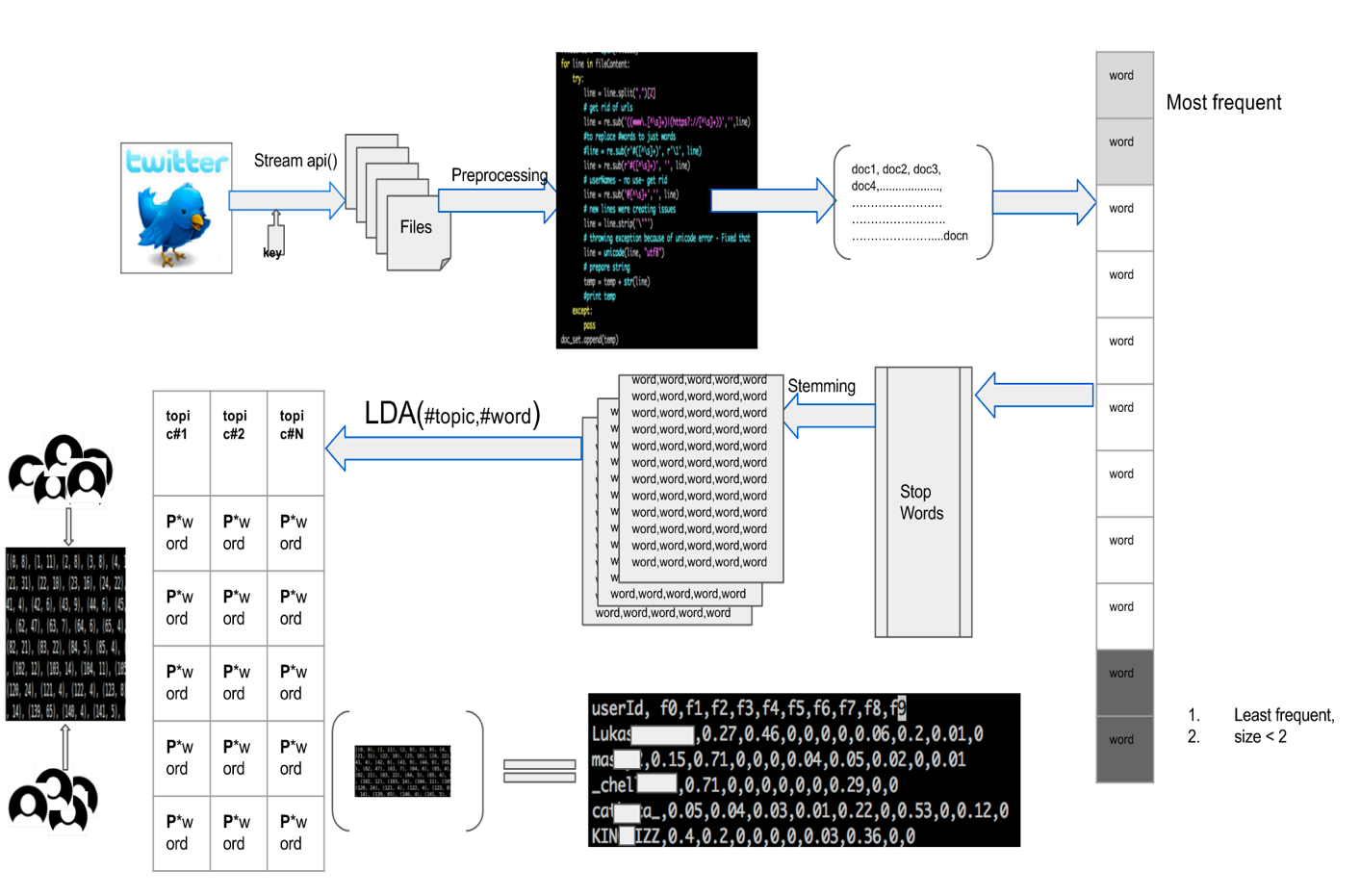


Figure High Level System Architecture for generating user feature vector

This pipeline illustrates how do we calculate user vector based on correlation with topic generated by LDA model.

High level of how Sentiment Analysis and other network feature works:

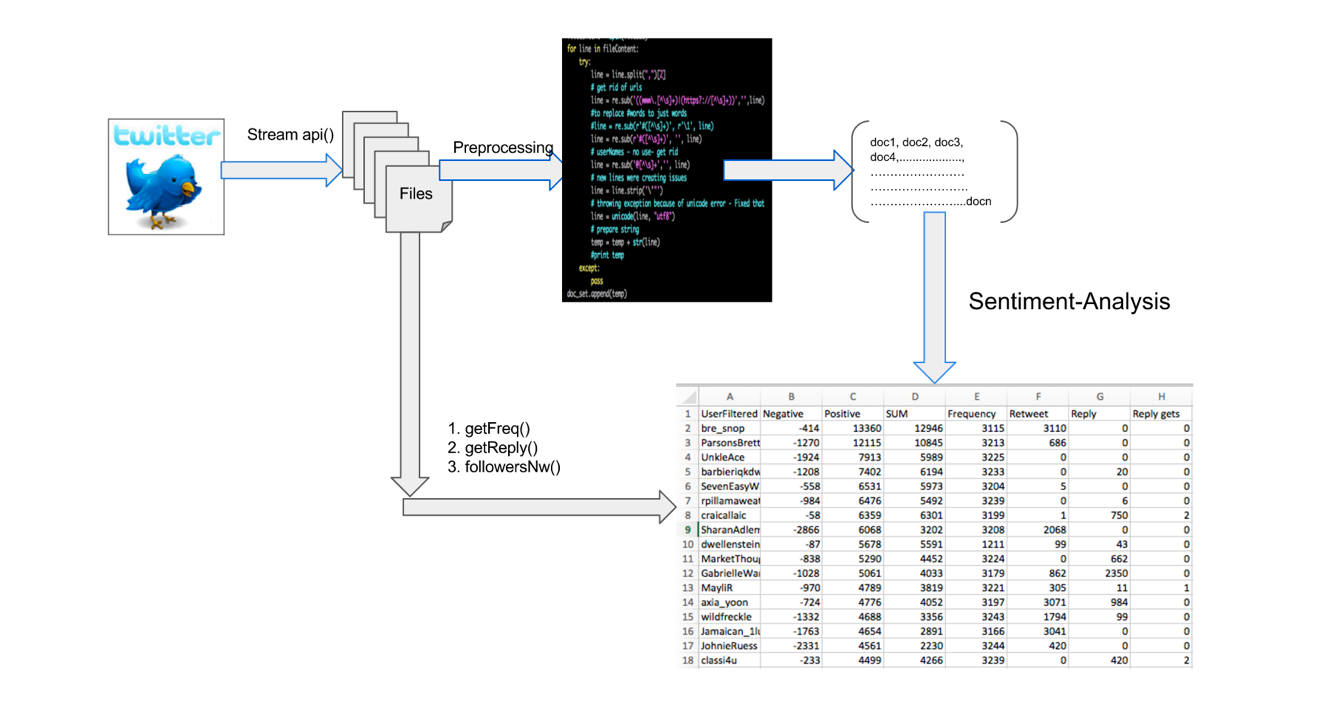


Figure High level system architecture for generating a feature vector in terms of user twitter data properties

**3.1 Data Collection**

All the data we obtain is public, and posted between 2014 and 2015 and was made available from Twitter via their APIs. At any point in time we did **not** made any attempt to obtain data which was either marked private or shared via direct message.

We used cloud virtual machine, Droplets by Digital Ocean [34] to download most of the twitter data. Some time when we needed more machine in parallel to work, we used Amazon’s EC2 [35] as well. Twitter provides two ways to access data. Because we wanted the data not to be biased, we went with stream option where we get access to 1% of public stream data without any filter.

Data downloading was mostly divided in 4 phases. Step 1 was achieved using stream API access where as 2,3 and 4 was achieved using REST APIs.

1. Connect to Twitter stream to download tweets from 1% sample access provided by twitter for free. We did this between Nov-Dec 2015. We then filtered user based on their language preference set as English (en).
2. At this stage we had more than 400,000 users. We just selected 10,000 of them randomly and downloaded their last available tweets. Twitter allows developers to have access to last 3200 tweets for a given userid (userid is identity of user on twitter and it has nothing of do with user’s name or any demographic info, it’s just a name which user have chooses to represent her/him on twitter network), using their APIs. We had then data for almost 9000 users as for some of the filtered users had private data policy set.
3. Out of 9000 users, we ran a script to remove users who had less than 100 tweets. The number 100 was selected based on purely experimental basis. We were left with close to 8000 users after applying this filter.
4. For these 8000 users, we also wanted to look at their friend network. Hence we used 6 EC2 machines to download their friends. This was very time consuming as users tend to have many friends. We later restricted the number to 5000 maximum friends for a given userid.

**3.2 Pre-processing**

Extracted tweets were in Java Script Object Notation (JSON) format. We extracted information we needed from this JSON object.



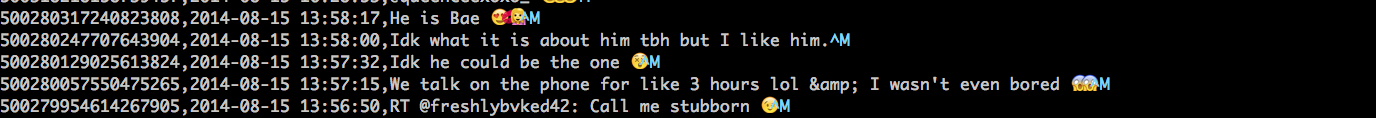
Figure Typical JSON format downloaded from twitter

As we can see that a typical tweet has lot of metadata associated with it. However, we were mostly interested in few very basic ones. At first just to get the user who had set English (en) as their language.

1. **def** extractTweet(inputdata):
2. #count = 0
3. **for** line **in** inputdata:
4. resultDict = json.loads(line)
5. **try**:
6. lan = resultDict["user”]["lang"]
7. **if** lan == "en":
8. var = resultDict["user”]["screen\_name"]
9. var = make\_unicode(var)
10. **print** (var)
11. **except** KeyError:
12. **pass**
13. **except** ValueError:
14. **pass**
15. **except**:
16. **pass**
18. **def** make\_unicode(input):
19. **if** type(input) != unicode:
20. input =  input.decode('utf-8')
21. **return** input
22. **else**:
23. **return** input

Once we had list of these users, we started downloading their last available tweets. For that we used tweepy [36]. We just downloaded in format {id, created\_at, text} format

e.g.



once we had a file for each user with this data, we were all set to perform sentiment analysis. These were the basic preprocessing steps we ran on these texts.

1. Remove the part of tweet’s text which had universal resource locators (urls) i.e. https or https
2. Remove hashtags (#), userids (@string), RT, RT”
3. Remove unicode characters
4. Remove the stopwords
5. Convert to lowercase

**3.3 Finding User’s Features using Twitter’s traditional data properties**

After preprocessing, we used TextBlob API to get sentiment score. TextBlob APIs call

produces one sentiment score for a given text. Modifying the source code for it to work on each word as text was overhead and was making program very slow.  Additionally, TextBlob uses inbuilt dictionary for scoring, hence it was not giving any weightage to slangs, shortcuts, and misspelled words which are very common in context of Twitter. We tried to fix this problem by trying to correct the misspelled words so we could use sentiment scores provided by TextBlob. However, experiment on sample data showed that, TextBlob.correct module was extremely slow. It took on an average two and half minute to process one document (one document is one user’s entire tweet text corpus). Keeping that number in mind we would have needed 2.30 minute each for 4000 users, which is approximately 10000 minutes, which is approximately 150 hours. Even with 3 machine working in parallel it was taking 2 days for one run. And we wanted to run it for all the possible topic counts like, 10, 20, 30, 40, and 50.  Hence we went ahead with our own sentiment analysis code. We used a simple algorithm which iterates over each tweet of a given user, after doing the preprocessing step, it tokenizes the words and then looks into to dictionary item for the score associated with the found word. It then based on sign of number (positive integer for positive word, negative integers for negative words) add the numbers to positive score or negative score.  This is a bag of word based model to compute the positive and negative score for each user. For creating a rich set of word and sentiment score we merged sentiment scores from two different sources to create a single word to score file. The negative word and files were:

1. AFINN-111.txt [27]
2. Negative-word.txt [36]

When a word was not found in dictionary, we gave it a zero (neutral) score. We cross validated it with TextBlob’s sentiment module. We were getting 13 out of 20 users same in the top negative list. We plotted these top 20% users to get a sense of trend.

**3.3.1 Users who fell in top 20% category of sharing negative content**

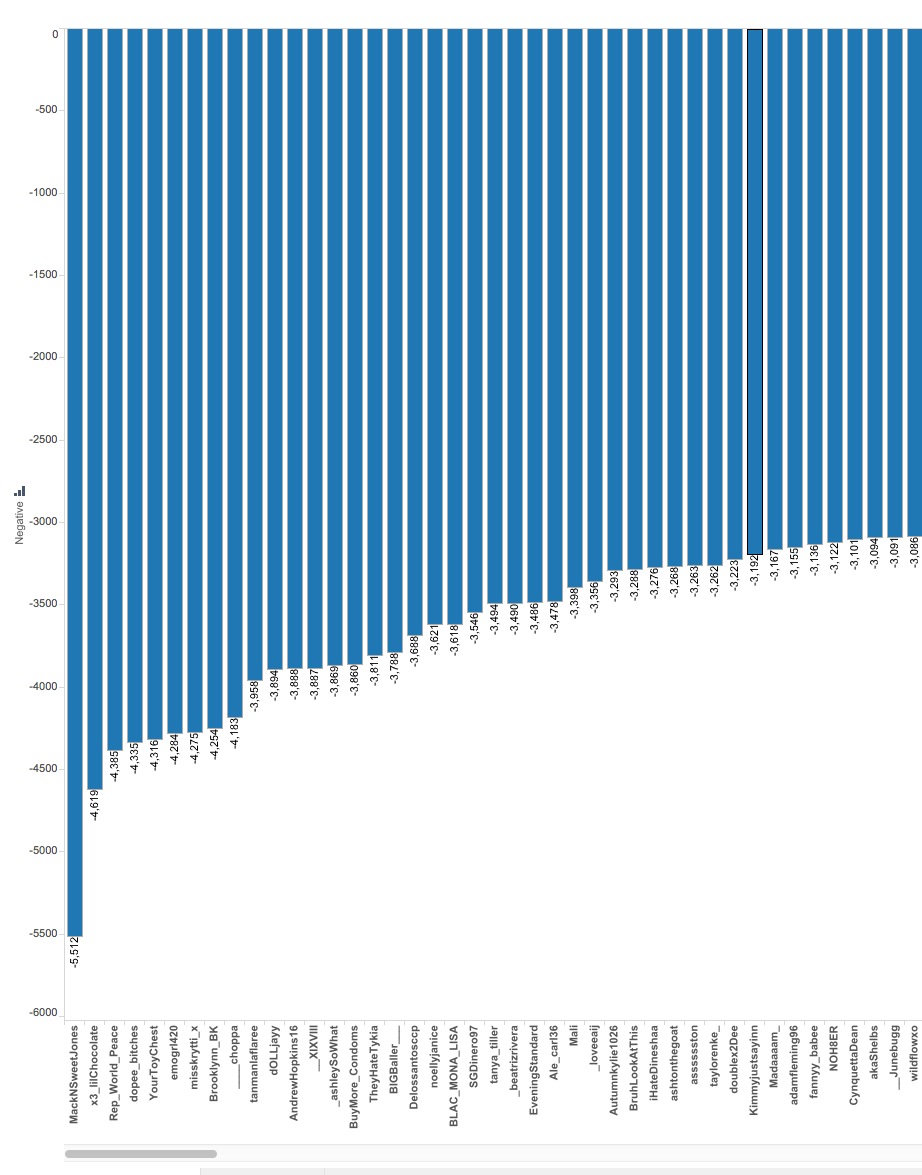


Figure User Sorted on Positive score

We also plotted users to whose negative contribution to social media was more than positive ones, i.e. the graph of user sorted on total score (positive + negative) .

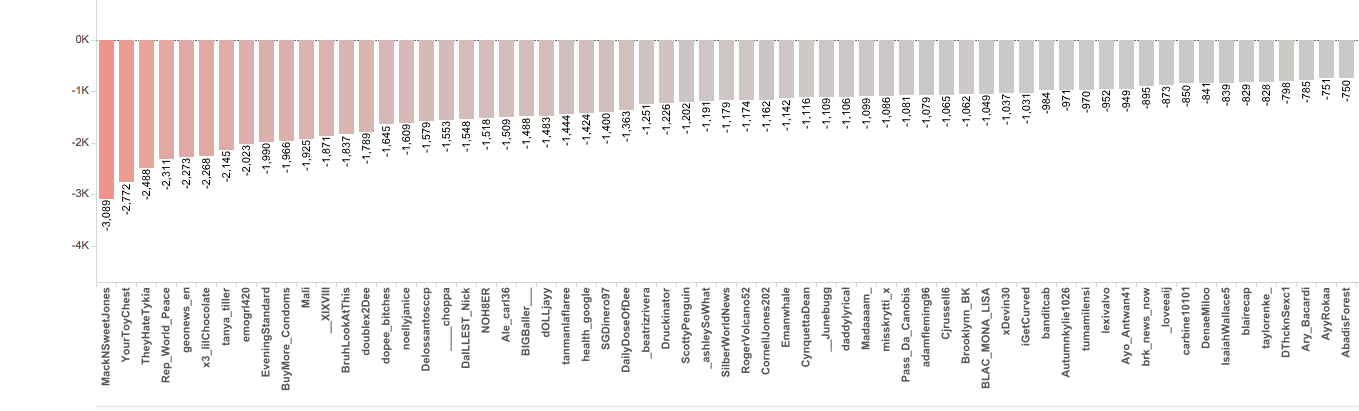


Figure User sorted on total score( negative + positive)

**3.3.2 Positive to Negative sentiment correlation**

We also looked at the distribution of positive to negative score across users to figure out how the distribution.

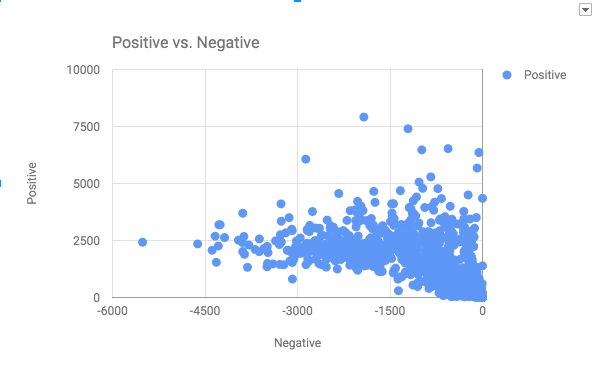


Figure Positive and Negative distribution of scores across 1k sampled users

Distribution was showing a clear trend that some users were polarized to share negative content [between -4500 to -3000] predominantly, but also had positive sentiment score of 1000-2500. However, users with high positive sentiment (greater than 5000) were sharing very less negative content.

**3.3.3 Sentiment correlation with “reply user does” and “gets”**

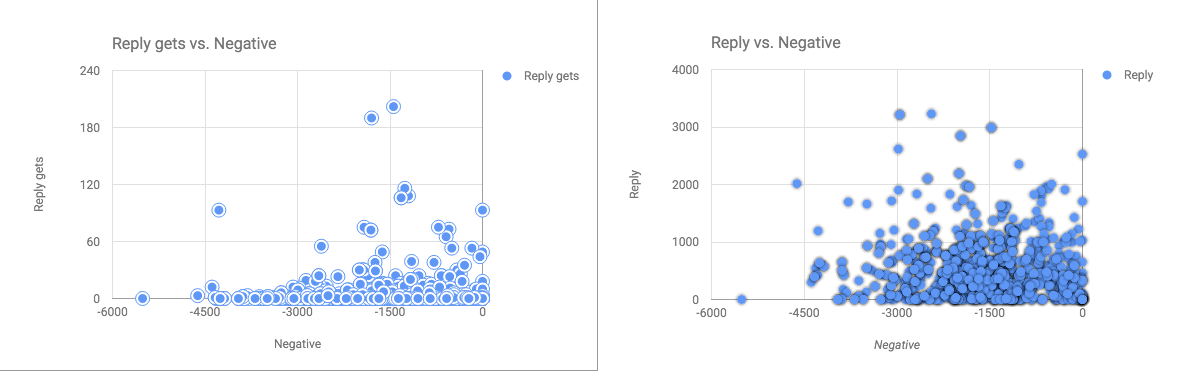


Figure Negative to Reply and Negative to reply gets

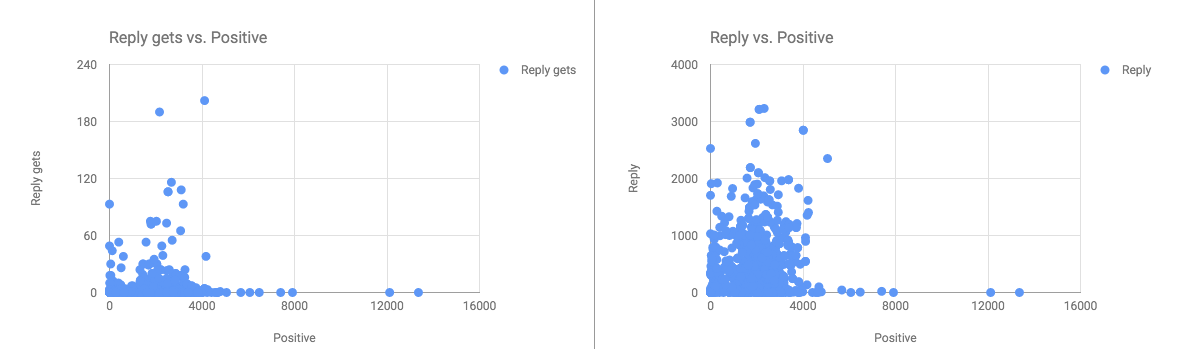


Figure Positive to reply and reply gets

Above graphs suggest that top 1% of negative user tend to reply lot more than top 1% positive users. This is a good signal to utilize as it seems user who have high negative score, get involve in communication with network users.

**3.3.4 Sentiment correlation with “Retweet Count”**

Next we plotted Below average negative to retweet count and above average positive to retweet count as well to see if there was something obviously standing out as trend.

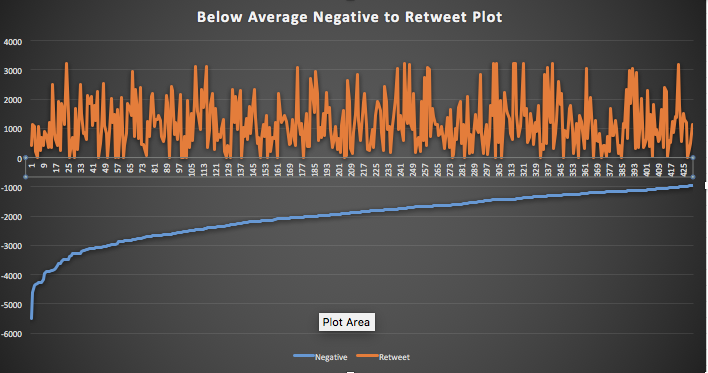


Figure Below Average Negative Sentiment to Retweet Count

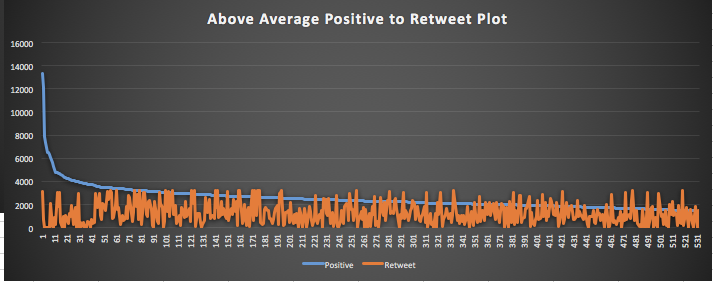


Figure Above average Positive Sentiment to Retweet Count

The plotted graph suggested that there wasn’t any pattern in the data we had for experiment. So we stopped pursuing this for feature selection.

**3.3.5 Sentiment correlation with “correctness (spelling) of tweets”**

However, there were few interesting patterns about sentiment to sentence correctness factor. We found it quite fascinating that there was obvious link between sentence correctness to negative sentiment.

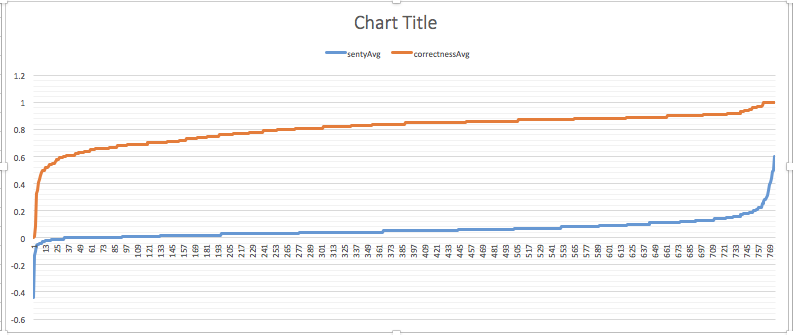


Figure Sentiment to Correctness Correlation

The graph seems to be suggestive of sentiment score directly proportional to correctness of the tweet. That suggest that most positive tweets are the well formatted ones and have very less spelling mistake. However, the most negative ones are also the least correct ones.

**3.3.6 Sentiment correlation with “Choice of words”**

We also investigated the idea if “choice of word” was different for people who were mostly tweeting negative in comparison to random people. For this after doing preprocessing steps on tweets, we ran a simple wordcount program to look at the frequency of words.



Figure WordCloud for “Word Choice” of 20 most negative users

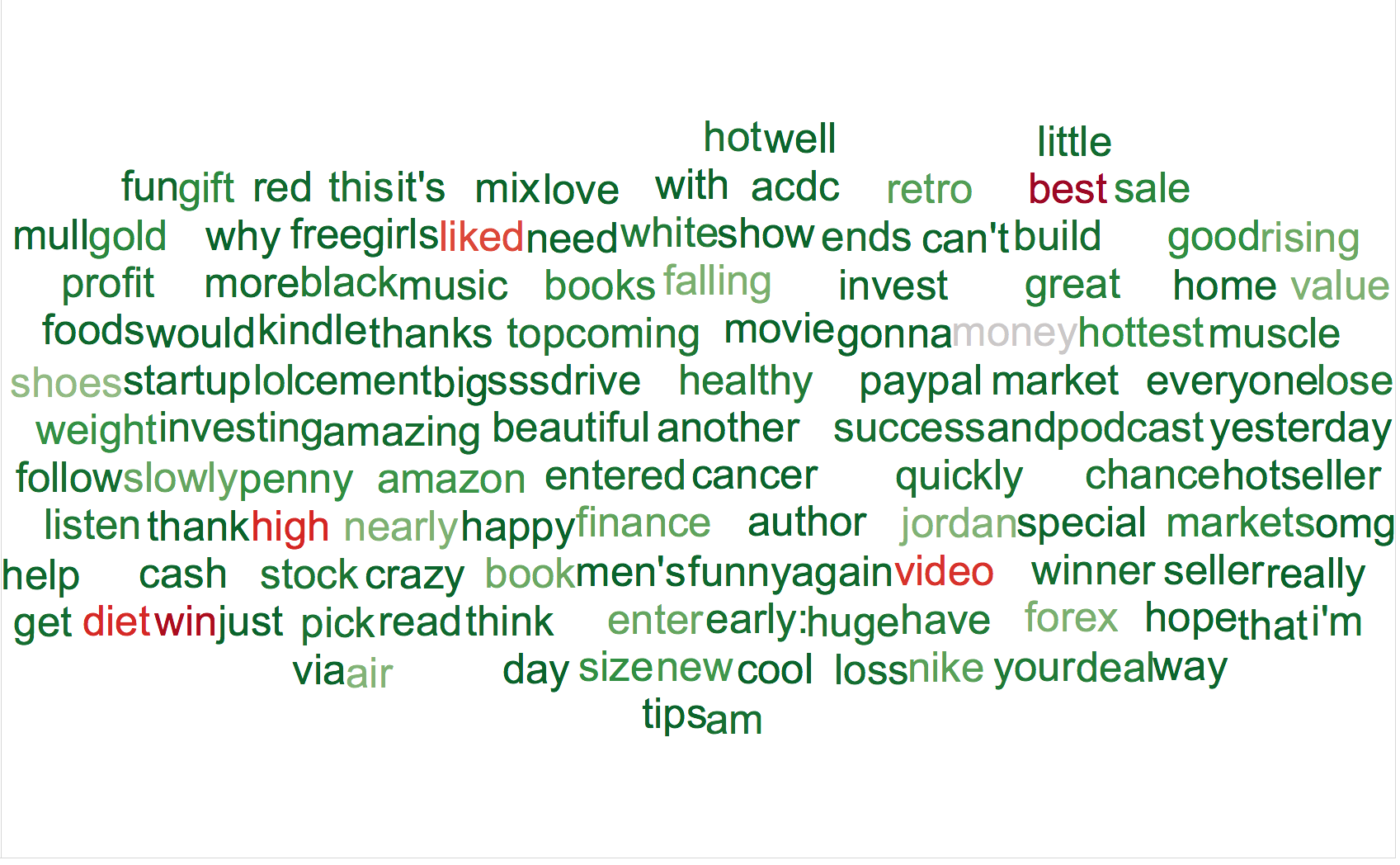


Figure WordCloud for “Word Choice” of 20 most positive users

**3.3.7 User’s network structure**

We also looked at the user’s network structure to catch some strong differentiating network substructure between negative and positive subcategory. We dominantly used Gephi for that purpose. However, we quickly realized that, most of the users whose negative sentiment was dominant, were mostly either “comedy host shows” or “sports club” ids. We downloaded all the friends for userids and created a adjacency matrix representation for it.

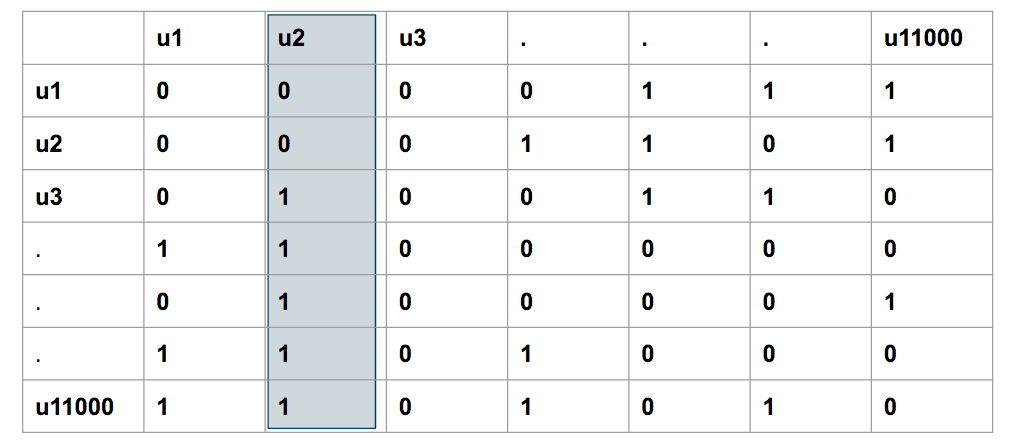


Figure Adjacency matrix representation of user and their friend network

To understand this matrix, a value of 1 at u3,2 represents that u3 has u2 in her/his friend’s list.The idea behind this graph was to find if there were “in-network” celebrities and if their content is affecting the network negativity or positivity over all.

When we plotted this matrix using tool Gephi [37], we got a graph like pasted below. Here, every user is represented by a node and size of the node is proportional to indegree of that node. Also, every edge represents a direct relationship between two nodes.

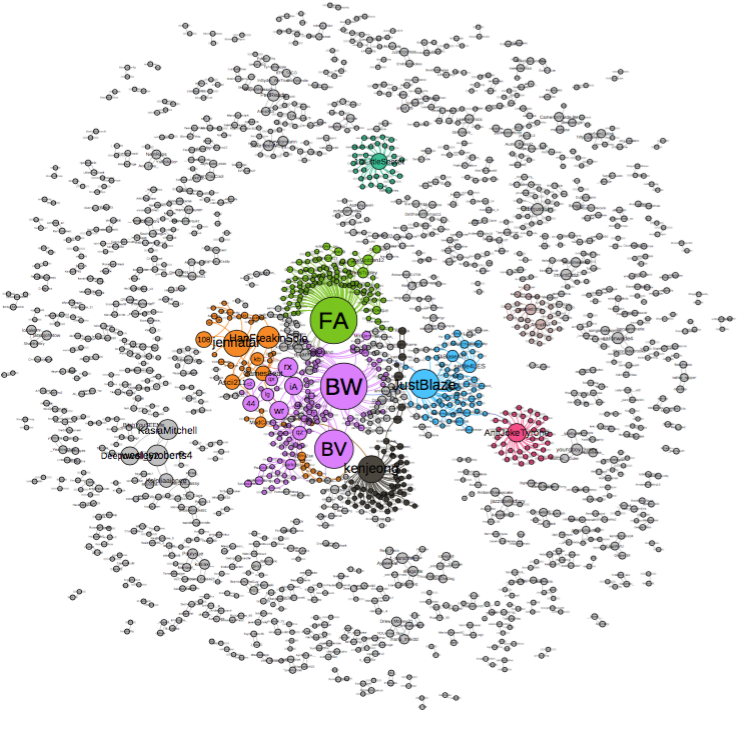


Figure network graph of users and their friends inside network

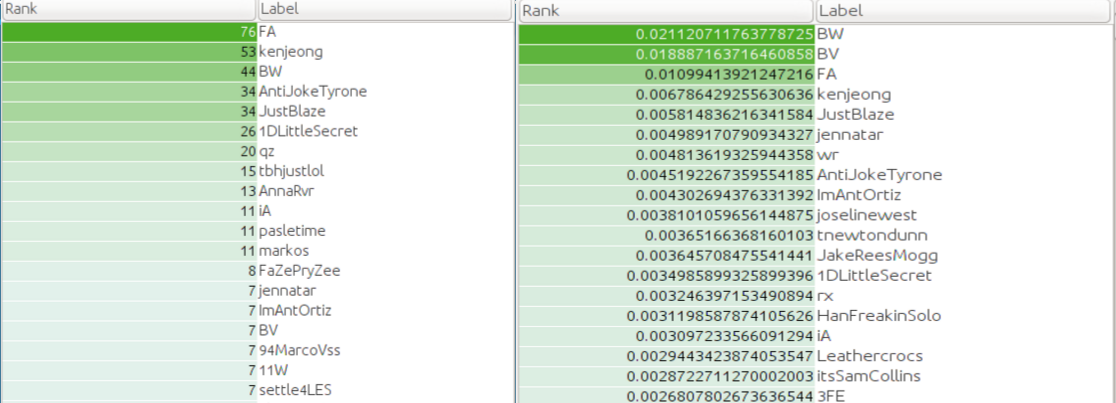


Figure Users in network sorted based on their indegree and pagerank

We did basic manual analysis on users and figured that the users were mostly either comedy shows official page or “one line offensive jokes”. Because most of such ids tweet at a very high frequency, it was very natural for these userids to have high negative sentiment.

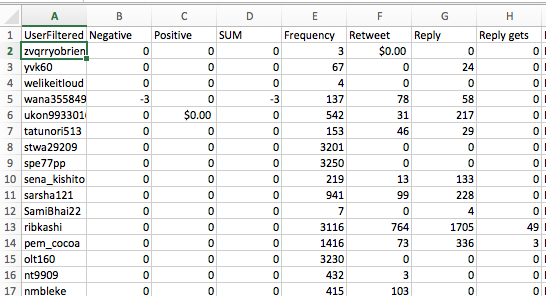


Figure Feature vector of each user based on analysis

On this feature vector data, we tried to run EM, an unsupervised clustering algorithm.

The expectation was that it should be able to cluster the most negative users in one group.

EM run on this dataset gave us 7 possible clusters.

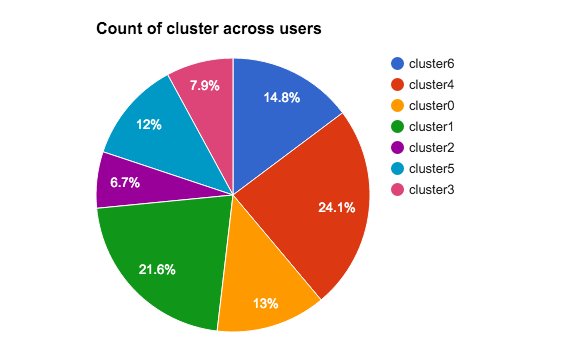


Figure Distribution of Clusters across dataset

However, most of the top 20 most negative users based on negative sentiment score were clustered in single group.

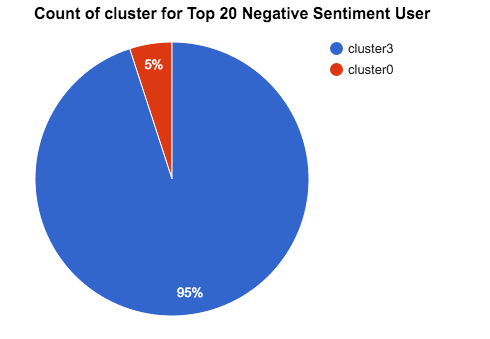


Figure Cluster Distribution for Top 20 Users

This was good result, as EM clustering algorithm was able to find correlation between all negative users. So we ran another experiment by tagging all users who had above average negative sentiment score as negative user and tried to see how the cluster distribution looks like.

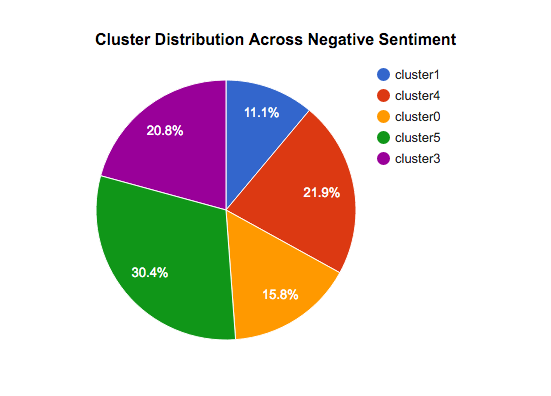


Figure Cluster Distribution of all users marked as Negative

This result was not very encouraging as the negative users were all over the cluster.

However, as one might expect, the results are heavily dependent on sentiment score. For the validation of results, we had two obvious options in mind. Either manual tagging or crowdsourcing. Both of them highly impractical and non scalable.

The negative sentiment produced by these were outnumbering individual negative comments. Hence we stopped looking at these features temporarily.

At the end of this “Traditional Twitter data analysis” we had feature set as mentioned below.

1. Negative sentiment
2. Positive sentiment

**3.1 Disaster Management using social media**

One of the research projects where Twitter is used as a source for event detection is LITMUS [(Tarchi, Casagli et al. 200](#h.3as4poj)[3](#_ENREF_27)) which combines physical sources and social sources for the detection of landslides occurring in the environment. Twitter was used as the social source where, tweets containing keywords like landslide, mudslide, rockslide etc. were extracted. The paper has highlighted two challenges that are commonly associated with the data coming from social media like Twitter namely, large amount of noisy information and unavailability of sufficient geo-tagged tweets. The noisy data consist of tweets that are not associated with landslide which cause the data extracted to be inefficient. One of the possible way to remove these noisy tweets, put forth by the authors of this paper is the machine learning technique called text classification using augmented ESA (Explicit Semantic Analysis[)(Gabrilovich and Markovitch 200](#h.3rdcrjn)[7](#_ENREF_11)). They have improvised the traditional ESA technique which is time inefficient, by applying K-means clustering algorithm, which will form clusters of similar items. This augmented ESA technique proposed by [(Tarchi, Casagli et al. 200](#h.3as4poj)[3](#_ENREF_27)) parse only partial Wikipedia documents based on the top N terms obtained from each cluster. This technique helped to classify the tweets into relevant and irrelevant categories more efficiently and which eventually helped in better event detection by considering only the relevant ones. They have also studied a technique for geo-tagging tweets whose geolocations are not provided by the users. They have made use of the NER (Named Entity Recognition) to assign geo-locations based on the location mention in the tweets. They increased the efficiency of NER by clustering the tweets using semantic distance and if a cluster had an overwhelming geolocation, then all the tweets in that cluster where assigned with this overwhelming geolocation. Therefore, this research has proposed techniques to overcome the two major challenges of social media data by improvising the existing techniques. This filtered data in addition to the data obtained from physical sources gives better results in the process of event detection.

Sakaki et al have implemented a real-time earthquake detection technique using tweets posted by the Japanese users on Twitter on the frequent earthquakes happening in Japan. They claim that earthquake events are posted on twitter much before media or USGS reports them. Though this research study is currently carried out on the earthquakes occurring in Japan, but they claim that this technique can be used to detect any other event occurring in any part of the world. If an earthquake is detected by this system, they send an alert email to all the registered users to take necessary precautionary measures and reduce casualties. They have extracted tweets containing keywords like earthquake and shaking for the analysis. They classified the tweets into two categories, positive class and negative class based on whether the tweet is referring to an actual earthquake occurrence or not. They have used SVM (Support Vector Machine) machine learning algorithm for the above classification. The tweets from the negative class are not considered for the analysis. Their system consists of two probabilistic models for event detection namely, temporal model that uses the time of the tweet being posted for predicting the earthquake occurrence at other locations and, spatio model that uses the location information from the tweets to predict where the earthquake is actually taking place. For detecting an event, they have considered a threshold value. If the probabilities calculated by the above two models exceeds this threshold value, the event is detected. Then using the location information, alert emails are sent to the registered users. For JMA (Japan Meteorological Agency) seismic intensity value greater than 3, their system detected 96% of the earthquakes. In this way, this research project takes the advantage of the real time nature of twitter and helps in the early detection of earthquake [(Sakaki, Okazaki et al. 201](#h.qsh70q)[0](#_ENREF_26)).

Bowden et al. have also implemented an earthquake detection system by keeping a track of the number of tweets extracted per minute containing the keyword earthquake and its possible usage in other languages like gempa, temblor, terremoto and sismo. The system detects any sudden increase in the frequency of tweets consisting of the above keywords. They proposed that any sudden increase in the number of tweets indicates that an earthquake has been detected. This increase in frequency of the tweets is detected by the short term average over long term average algorithm usually used in seismology. “Detection time is very fast, with about 75% coming in before two minutes.” [(Earle, Bowden et al. 201](#h.17dp8vu)[2](#_ENREF_10)). From this we can see that, the real time nature of Twitter helps in event detection with good efficiency [(Earle, Bowden et al. 201](#h.17dp8vu)[2](#_ENREF_10)).

Another contribution towards event detection is for the natural disaster of fire. Power et al. have implemented a system that tracks real-time tweets describing fire related events happening in the vicinity. They have extracted tweets containing keywords fire or smoke and then carried out text classification to find out tweets related only to the fire breakout incident. They also collected tweets from the users who live in the vicinity of the affected area. This system allows user to select the desire location from the map provided and once the region is selected, and an incident name is provided, the system gives all tweets related to fire or smoke from the selected region. It has an option of text classification that gives tweets related to fire only. Also it provides a tweets per minute count to detect any sudden spike in the number of tweets obtained. The system also allows to simultaneously monitor up to four fire events as also monitor events happened in the past that are saved as event of interest [(Power, Robinson et al](#h.2xcytpi)[.](#_ENREF_22)).

Power et al. have developed a similar system for detecting fire related events in addition to the alert or notification which is sent in the form of an email giving the level of intensity of the alert. In order to send efficient and helpful notifications, they filtered out tweets not describing fire related activities using SVM (Support Vector Machine) machine learning algorithm. They found out the frequency of the root words in the tweets related to fire event by performing various preprocessing steps like removing punctuation, performing stemming and assigned different alert levels depending on the intensity. Using the above preprocessing techniques, they came up with 17 root words related to fire and each word had a different alert level and threshold value. If the frequency of word crosses the defined threshold value, a notification email was sent with the necessary details. “In the first three months of operation, the system generated 42 ‘fire’ email notifications where only 20 corresponded to real fire events. Filtering these alerts using the classifier resulted in 21 notifications: an improvement in accuracy from 48% to 78%, albeit with a reduction in recall from 1 to

0.8” [(Power, Robinson et al. 201](#h.1ci93xb)[3](#_ENREF_23)).

In Tweet4act [(Roy Chowdhury, Imran et al. 201](#h.2bn6wsx)[3](#_ENREF_25)), tweets are classified depending on whether it talks about pre-incident, post incident or during incident conditions. They have collected tweets on three events namely, Joplin Tornedo, Haiti earthquake and Nesat Typhoon. Initially, they classify the tweets into categories depending on whether they are related to the event or not using k-medoid [(Hodge and Austin 200](#h.26in1rg)[4](#_ENREF_12)) based filtering algorithm. Then all the tweets that were classified as tweets related to the events were further classified in to pre, during and post periods using dictionary based classification. They created a dictionary of words for each period and also used parts of speech to classify the tweets. Depending upon the words in the tweet, the three periods were assigned scores. The period with the maximum score was used to assign label to that tweet. The words were divided into the three periods depending on tense and parts of speech [(Roy Chowdhury, Imran et al. 201](#h.2bn6wsx)[3](#_ENREF_25)).

Acar et al. have studied the possible tweets coming from disaster affected area and tweets coming from indirectly affected area. The tweets posted by users who directly suffered from the disaster talked more about their safety and other survival related situations. Whereas people from indirectly affected areas talked about the post effects of the disaster in terms of environment, industries, transportation etc. Also the paper talks about some features of Twitter that are not reliable for posting tweets related to disasters and have put forth a few features as suggestions that should be inculcated in Twitter which might help improve in analyzing disasters more efficiently. These suggestions include making official hashtags for disasters, limiting the number of RTs per hashtag etc. to reduce the spread of false information among the general public. Also the paper throws light on how the reliability of Twitter is at question and how people find it difficult to trust a tweet seeking help. Due to this, many people who were actually seeking aid received no reply. But still the authors have the paper have claimed that out of all the possible sources of communication, Twitter was the most reliable one [(Acar and Muraki 201](#h.1fob9te)[1](#_ENREF_2)).

Palen et al. have used Twitter as a tool to study the flood related activities on Social Media during the flood warnings declared on the Red River in USA, which flows into Canada. During the period of flood warning, tweets containing the keywords red river and redriver were extracted. From these tweets they found the unique users posting the tweets and started following their activity on twitter. They found a sudden increase in the number of tweets being posted in the affected areas of floods and most of the tweets were describing about the post condition of the floods. Out of the unique users followed, some of the users were found volunteering in the flood affected areas and were also asking for help. This created awareness among the general masses to volunteer thus helping reduce the impact. They also found some of the users opting to stay in the flood affected areas to analyze the present condition and post tweets regarding the river level and other minute details seen at the affected area. The paper also describe that retweets about disasters can be used as an indicator of trustworthiness of the tweet. So they have concluded that retweets from local residents of the affected area can be used as source of dissemination of information, which is true. Thus they have described social media as a platform for seeking help and spreading real time alerts and information about the disaster affected area [(Palen, Starbird et al. 201](#h.1y810tw)[0](#_ENREF_20)).

Chatfield et al. have described how Twitter can be used as a source of fast information propagation for issuing early disaster warnings. “The Indonesian government issued its tsunami early warning Tweet, which was re-tweeted without delay by its followers to their own followers to warn tsunami hazards during the 2012 earthquake. Within 15 min it reached over 4 million Twitter users.[”(Chatfield, Scholl et al. 201](#h.3dy6vkm)[3](#_ENREF_6)). Indonesian Agency for Meteorology, Climatology and Geophysics’ (BMKG) Twitter account was used to propagate Tsunami related early warnings to its followers and which was eventually re-tweeted by their corresponding followers. The paper shows that once the Early Tsunami warnings were posted, within few minutes there was a sudden increase in the number of tweets talking about this warning and there were followers concurrently retweeting about it. These early warning tsunami tweets gave almost 9 minutes for the residents of the coastal areas to evacuate themselves. The expected Tsunami timings issued where based on the arrival times of the earthquake in the stated areas. The paper states that one of the reason for the faster propagation of the early warnings of Tsunami were the followers of the BMKG twitter network which consisted of renowned news channels who had retweeted the warnings to larger audience [(Chatfield, Scholl et al. 201](#h.3dy6vkm)[3](#_ENREF_6)).

**3.2 Disease monitoring using social media**

Social media is not only used for detecting disasters in the environment but also used in medical field to reduce the after effects of seasonal epidemics. “Twitter data provides real time assessment of Influenza-like-illness (ILI) activity.” [(Achrekar, Gandhe et al. 201](#h.3znysh7)[1](#_ENREF_3)) . Using the real-time tweets, they tried to keep a track of new users reporting flu like terms in their posts. They found that the number of tweets reporting influenza like illness were closely related to the number of influenza cases reported by CDC (Center for Disease Control and Prevention) with a Pearson correlation coefficient of 0.9846. They also tried to predict the level of ILI activity in terms of percentage of visits to the physician for the successive weeks, by using the CDC data from the previous week and combining this data with the real time tweets on flu related terms from Twitter. They have also highlighted that the use of tweets in the regression model to predict the number of visits to the physicians in the successive weeks improved the efficiency, as compared to the regression model without the tweets. This shows that Twitter could also be used as the source for tracking and predicting seasonal epidemics like H1N1 influenza flu activity in the general public thus causing the general public to take precautionary measures and preparing themselves for any large scale epidemic.

Corley et al. have made use of web and social media for detecting influenza-like illness activities. They have collected blogs that contain the two keywords flu and influenza. The results of the analysis performed shows a strong resemblance to the flu season that had started in USA in 2008. The blogs containing the two keywords were collected for a period of two months from 1st August 2008 to 30th September 2008, which was the start of the flu season in USA. They have calculated the number of blogs obtained every week, month and day of the month. “The day of week Flu Content-post frequency is averaged for August and September and is normalized by the corresponding day of week average for all blog-world posts, this data is plotted” [(Corley, Mikler et al. 200](#h.1t3h5sf)[9](#_ENREF_7)). From this normalized Flu-Content post frequency, it was observed that the number of posts in August is stable as compared to the sudden increase in the number of posts in September which is assumed to be the starting month for the flu season. One of the Influenza like illness activity given by the Healthcare providers on the number of physician visits and FC-related post show a Pearson co-relation of 0.767 for the day of week window of 21st September 2008. The paper has also contributed significantly by finding frequent bloggers or flu-related communities that could help spread the awareness of flu in case of an outbreak. In this way, social media in the form of Weblogs can also be used for the analysis and detection of flu-like epidemics [(Corley, Mikler et al. 200](#h.1t3h5sf)[9](#_ENREF_7)).

Paul et al. have analyzed general public health concerns using Twitter as a social media. They have used Twitter as a platform to study their Ailment Topic Assessment model on tweets talking about health related problems like allergies, obesity, insomnia which had not been analyzed earlier. Though these problems are not severe as epidemics but can have severe consequences over the period of time [(Paul and Dredze 201](#h.4i7ojhp)[1](#_ENREF_21)).

CHAPTER 4

SYSTEM ARCHITECTURE AND OVERVIEW

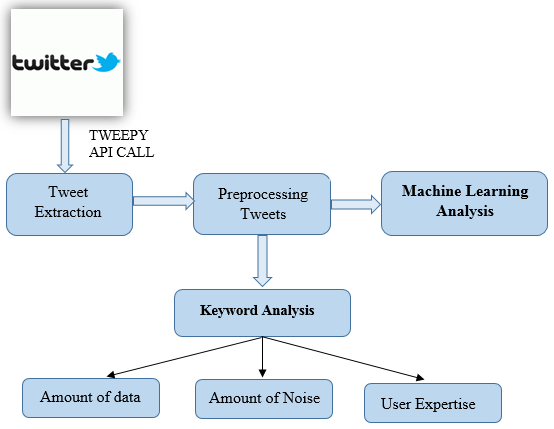
In this chapter, we have described the challenges faced while performing the study of CyanoHABs using Twitter and also described the System Architecture in detail. We have also talked about the data extraction technique and the number of tweets extracted for the keywords used for analysis.

**4.1 Challenges:**

While using Twitter for the analysis and monitoring of CyanoHABs, we encountered a few inherent challenges:

1. There are certain phenomena that are associated with multiple terms or hashtags. CyanoHABs are also known by multiple keywords like Algae Bloom, Blue Green Algae, Red Tide, #CyanoHABs, #cyanobacteria, #microsystin. Amongst them, some keywords (cyanobacteria, microcystin) are used by people who are experts in this field like domain scientists, PhD or research institutes. Whereas some keywords like (algae blooms, red tide) are commonly used by general public. The number of tweets containing technical terms is less as compared to the number of tweets containing generic keywords. Therefore in such a dichotomy, the challenge is to determine which keywords should be used for data analysis, technical terms or generic terms?
2. The second challenge that we came across was to do with the reliability of the posts on Twitter. The posts by general public or non-experts would have less level of reliability or trustworthiness as they may or may not have expertise in the field and so may not be sure about the identification of CyanoHABs on water bodies, though it will be still considered as a piece of information. This might also introduce noisy data. On the other hand, the level of reliability and trustworthiness is more when the post comes from an expert in the field resulting in more specific and correct information, thus reducing the amount of noise.
3. The third challenge is associated with the use of machine learning algorithms to classify the data into relevant and irrelevant ones. A training model is generated from a set of labeled tweets and this model is used to classify all newly obtained data. But our research highlights the shortcoming of this “train once classify ever” model on social media data, where same training model containing data from the past is used to classify present data. We have studied the performance of “train once classify ever” machine learning technique, when the periodic gap between the training model and newly obtained data increases. This might result in incorrect classification of data, which will eventually hamper any analysis made on it. The reason behind this being that social media data evolves with time and has changing characteristics. Hence, classification using machine learning techniques becomes inefficient because of the less similar events in training and testing datasets. Thus, using machine learning techniques to classify social media data cannot remove human involvement totally. Our experimental results show that instead of manually labeling all the new data, every time for retraining, one of the possible way could be to only use a fraction of data or tweets from each month and retrain the model to get the desired classification of data with better accuracy.

**4.2 System Architecture**



Number of users

*Figure 2: System Architecture and Overview*

Figure 2 shows the system architecture and overview. It consists of three primary components namely,

* Tweet Extraction
* Preprocessing of Tweets
* Keyword and Machine Learning Analysis

The Tweet Extraction component consists of extracting tweets from Twitter using the Tweepy [(Roesslein 200](#h.3whwml4)[9](#_ENREF_24)) API. It collects all the tweets for the given keywords. For our research, we have collected the tweets containing keywords related to CyanoHABs namely Red Tide, Algae Bloom, Cyanobacteria and Blue-Green Algae.

The preprocessing component implements the preprocessing techniques on the tweets such as removing special characters like #, @,! etc., removing Retweets i.e. RTs from the set of extracted tweets to avoid redundancy. Finally, all the tweets are then converted all to lowercase. The Keyword and Machine Learning Analysis are the main focus of our research, which are described in detail in the following chapters.

## **4.3 Data Collection**

In order to use social media for the surveillance of CyanoHABs growing on water bodies, we extracted tweets containing terms Cyanobacteria, Algae bloom, Red tide and Blue Green Algae. The tweets containing the above keywords were extracted for duration of seven months starting from September 2014 to April 2015.

*Table 1: Number of Tweets*

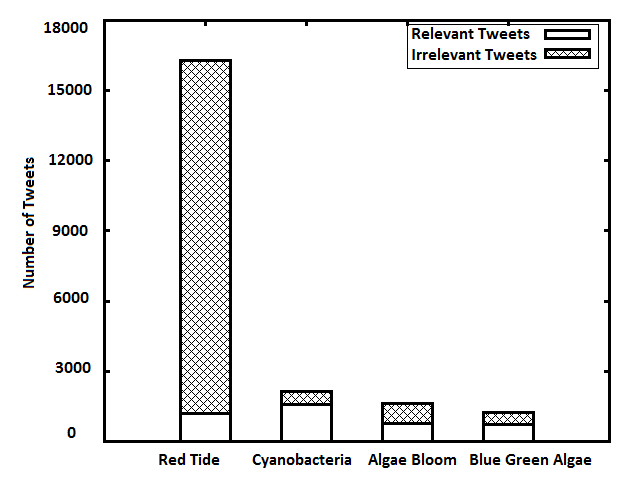
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Hashtag** | | | |
| **Months** | Red Tide | Cyanobacteria | Algae Bloom | Blue Green Algae |
| September | 3891 | 210 | 319 | - |
| October | 2274 | 330 | 148 | - |
| November | 3060 | 311 | 262 | - |
| December | 1894 | 91 | 164 | - |
| January | 1676 | 401 | 255 | - |
| February | 1027 | 501 | 263 | 487 |
| March | 2463 | 326 | 216 | 439 |
| April | - | - | - | 297 |
| Total | 16285 | 2170 | 1627 | 1223 |

However, for the keyword Blue Green Algae, the data extraction was started late from February 2015. Table 1 shows the number of tweets collected for each month for all thefour hashtags*.* The tweets that were extracted had a lot of noise, especially for those keywords that are associated with multiple events. For instance, the keyword red tide is also associated with a musical band, US political elections etc. Such tweets are being referred to as noise as they do not talk about CyanoHABs growing on the surface of water bodies.

CHAPTER 5

KEYWORD ANALYSIS

In order to determine the number of noisy tweets associated with each keyword, we asked a group of five human evaluators to label the tweets as relevant or irrelevant depending on whether the tweet actually talks about the algal bloom on water bodies or not. The human evaluators were given set of instructions for labeling which helped them assign labels to the tweets and categorize the tweets efficiently into relevant and irrelevant ones. Figure 3 shows the number of relevant and irrelevant tweets for each hashtag.



*Figure 3: Number of Relevant and Irrelevant Tweets for each hashtag*

In keyword analysis, we study the relationship of technicality of a term or hashtag in Twitter to:

1. Amount of data
2. Number of users
3. User Expertise.

In the following paragraphs, we have described each relationship with the technicality of terms in detail.

1. **Amount of data:**

As stated previously, out of the four keywords used for tweet extraction, tweets containing the keyword Blue Green Algae were collected for duration of 3 months and the other keywords namely Red Tide, Algae Bloom and Cyanobacteria, were collected for duration of 7 months. So in order to have a uniform base for comparison we have calculated the rate of relevant and irrelevant tweets for each hashtag by dividing the total number of relevant/irrelevant tweets by the number of months for which they were collected. The total rate is also determined by dividing the total number of tweets obtained by the number of months for which they were collected.

From the total rate, relevant rate and irrelevant rate, we determined the relevant percentage by dividing the relevant rate with the total rate, and determined the irrelevant percentage by dividing the irrelevant rate with the total rate. Table 2 shows the rate and percentage of relevant and irrelevant tweets.

From the above calculated rates and percentages, we can consider Red Tide to be a generic or common term because the number of tweets obtained for this keyword is the maximum which means that it is more popular among common people. But at the same

*Table 2: Rate and Percentage of Relevant and Irrelevant Tweets*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Total Rate | Relevant Rate | Irrelevant Rate | Relevant % | Irrelevant % |
| Red Tide | 2326 | 174 | 2152 | 7.48 | 92.5 |
| Algae bloom | 232 | 113 | 120 | 48.7 | 51.7 |
| Blue Green Algae | 482 | 282 | 200 | 58.5 | 41.4 |
| Cyanobacteria | 308 | 229 | 79 | 74.35 | 25.6 |

time, the relevant percentage is the least and it has the most number of irrelevant tweets. Hence we can say that for the common term red tide, which is known to a larger crowd, the total rate is maximum but the relevant percentage (R %) is the least because of the larger number of irrelevant or noisy data present in the extracted tweets.

Therefore for common terms, though the amount of information obtained is maximum, the relevancy rate is less, thus reducing the level of trustworthiness. Whereas for technical terms like Cyanobacteria and Blue green algae the relevant percentage is maximum but the total rate is less as compared to the total rate of Red Tide. This means that there is lesser noise and more relevant tweets in the total tweets extracted for technical terms, though the total number of tweets obtained are less. Hence the popularity of technical terms is less, since it will be known only to experts of this field, which increases the level of trustworthiness. We also see that the irrelevant percentage (I %) is gradually decreasing as the popularity of the keyword decreases.

1. **Number of Users:**

As stated previously, the popularity or number of users for commonly known terms will be more as compared to the terms that are technical. This hypothesis can be clearly

seen from the numbers in the Table 3. The Table gives the number of unique users for each hashtag.

*Table 3: Unique, Reliable and Active Users for each hashtag*

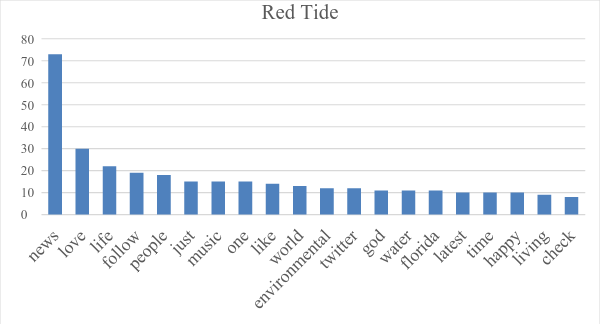
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Unique Users | Unique Users Rate | Reliable Users | Reliable User % | Reliable Users Rate | Active Users | Active Users Rate | Patron Users % |
| **Red Tide** | 10556 | 1508 | 958 | 9.05 | 137 | 126 | 18 | 13 |
| **Cyanobacteria** | 1295 | 185 | 914 | 70.5 | 131 | 160 | 23 | 17 |
| **Algae Bloom** | 1084 | 155 | 626 | 57.7 | 89 | 96 | 14 | 15 |
| **Blue Green Algae** | 934 | 311 | 532 | 56.91 | 177 | 94 | 31 | 18 |

The column Unique users gives the number of unique users posting tweets for each hashtag. The table also gives the number of Reliable users for each hashtag, which is the number of users posting relevant tweets. Active users give the number of users posting atleast two tweets. Since Blue Green Algae tweets were collected only for a period of 3 months, we determined the rate of unique users, rate of reliable users and rate of active users for each hashtag. These rates were determined by dividing the number of unique users, reliable users, and active users by the number of months for which they were obtained. Though Red Tide has more number of users, the Reliable User percentage is very less i.e. only 9% of the total users are reliable users that talk about the actual algal bloom growing on water bodies. This shows that the term red tide is more popular among general public but the number of contributors that provide relevant and useful information is very less. Whereas for technical terms, Cyanobacteria and Blue Green Algae, though the total number of users is less as compared to red tide, they have the maximum reliable user rate among the four keywords. Also, the Reliable user percentage is maximum for cyanobacteria from which we can infer that even though the total contributors are less, almost 70% are reliable contributors, which give relevant and useful information regarding the Algal Bloom growing on water bodies. From the above mentioned observations, we can infer that for technical terms like cyanobacteria, the majority of tweets might be coming from experts or domain scientists from this field, which are less in number as compared to common people. These smaller number of tweets will have valuable information and hence will have high level of trustworthiness because of the expertise of the people posting them. For generic terms we see that the number of contributors are more but out of them, the fraction of contributors that give actual information is very less. Also, since these generic terms are more popular among common people, who may or may not have expertise in this field, the level of trustworthiness eventually reduces.

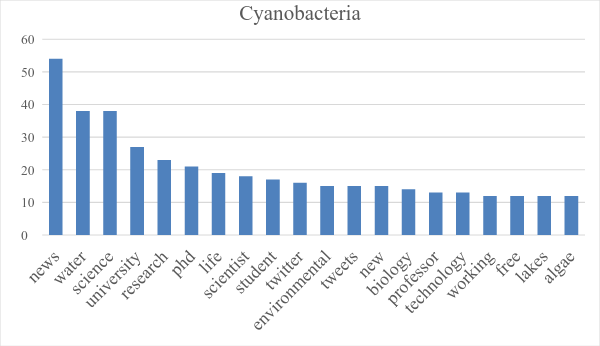
Table 3 also shows the Active Users % for each hashtag. Those users who post atleast two tweets on a particular keyword are considered to be active users. Active user rate is determined by diving the number of active users with the number of months for which the tweets for respective hashtag were collected. The Patron user % was obtained by dividing the Active user rate by the reliable user rate, which will state which hashtag is giving maximum information. Red Tide has minimum number of Patron Users % among the four hashtags. Whereas Blue Green Algae has maximum number of Patron User %. From this we can infer that for moderate technical terms, the amount of information obtained is maximum since the moderate level of technicality induces some level of trustworthiness and since it is mildly popular among generic people we can expect to have more number of tweets than a high level technical term. For high level technical terms like Cyanobacteria which is known only to a few experts, the number of contributors are less and the same set of users are found posting the posts on twitter. Hence we can infer that for technical terms, the frequent users are more as compared to common terms. Whereas for common words since it is popular among people, the number of frequent users are less and there are many users that tweeted only once.

1. **User Expertise**

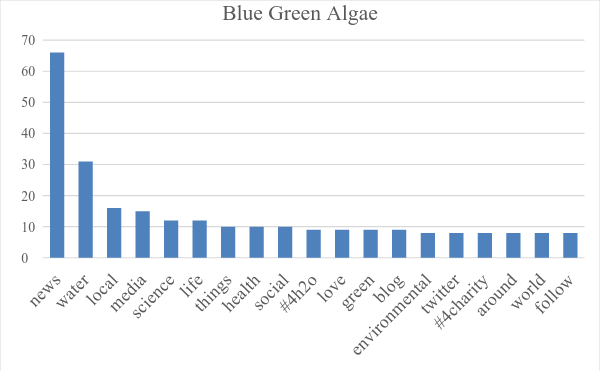
In order to find, what kind of users post tweets containing the above keywords, we analyzed the professional background of the users. The hypothesis that we had put forth was that more common or popular a keyword, more popular it will be among common people and more technical a term it will be known to lesser people who might be experts in the field.



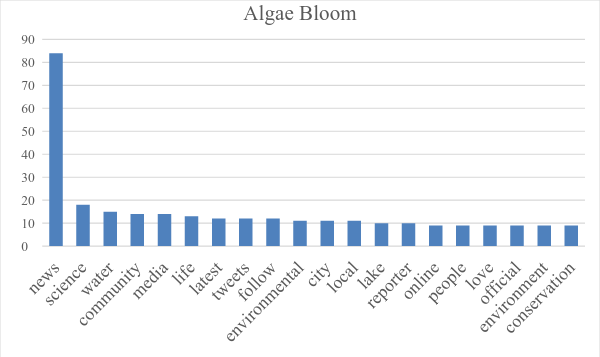
*Figure 4: Top-20 frequent words in the Description of Relevant Users (Red Tide)*



*Figure 5: Top-20 frequent words in the Description of Relevant Users (Cyanobacteria)*



*Figure 6: Top-20 frequent words in the Description of Relevant Users (Blue Green Algae)*



*Figure 7: Top-20 frequent words in the Description of Relevant Users (Algae Bloom)*

In order to support the hypothesis, the background or profession of the twitter needs to be analyzed. And this can be done with the help of the description provided by the users on their twitter profile. We used Twitter API, Tweepy [(Roesslein 200](#h.3whwml4)[9](#_ENREF_24)) to access the user information uploaded by the users on their profile.

We collected the description of the users who posted relevant tweets (i.e. Reliable Users) about algal bloom for all the four hashtags. Due to privacy concerns and misleading or hypothetical descriptions provided by majority of users it was difficult to conclude the profession of the twitter users. So we computed the word count of the descriptions obtained for each hashtag, which will give us an idea about the words appearing in the descriptions of the people posting tweets using these keywords.

Figures 4-7 show the top 20 frequent words appearing in the description of relevant users for each hashtag. We can see that the description of the users of the most technical term Cyanobacteria, had frequent words like PhD, scientist, professor, university, student, research etc. which shows this term is possibly used more by researchers, scientists or students dealing with the environment. From the frequent words of Blue Green Algae and Algae Bloom we can see that media reporters, blog writers or people concerned about environment or community, might more commonly use it. Whereas for red tide which is the most common term used for algal bloom, we see that the possible users are common people or general users of twitter, and have words like love, music, happy, people etc. as the frequent words in their description.

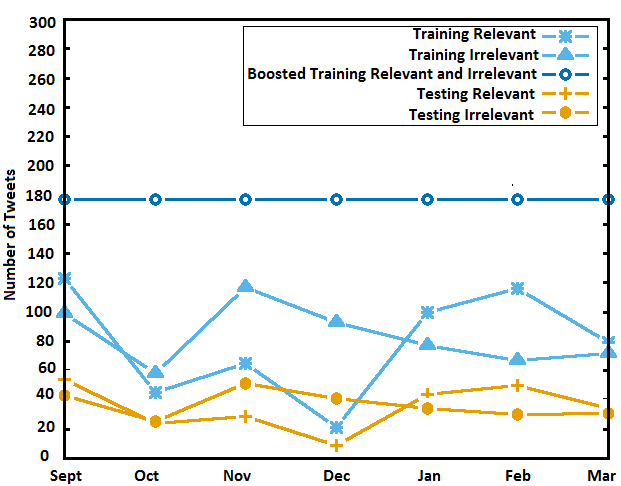
Therefore for technical words, the description of its users shows that these words might be more popular among researchers or scientists. Whereas for mildly technical terms like Blue Green Algae and Algae Bloom, these terms possibly are more popular among journalists, blog writers or people concerned about the environment. For generic terms like red tide, which could be more popular among general public, are mostly used by common people.

CHAPTER 6

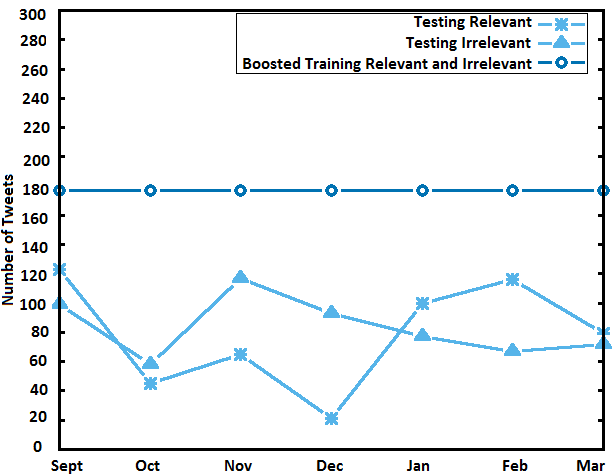
MACHINE LEARNING ANALYSIS

Twitter has real time streaming data associated with it. And this data has evolving or changing characteristics. For example, when an earthquake occurs in Japan, we can expect majority of the tweets on earthquake to talk about this disaster. The discussion will eventually fade out as the event becomes old or another earthquake occurs at a different location. Such is the evolving or changing characteristics of the data from social media. Another aspect of Twitter being the noisy data associated with it. As discussed in chapter 4, the amount of noisy data varies as per the popularity of the hashtag. This means that whenever we extract tweets on a particular keyword associated with a disaster, we get a collection of tweets that are related to the disaster as well as those that are not related to disaster. Therefore, while using social media for environmental monitoring it becomes necessary to filter out the irrelevant tweets from the relevant ones for better results. For smaller amount of data, human involvement becomes feasible where the humans could be instructed to assign the tweets as relevant or irrelevant depending on whether they are describing the disaster or not. But for large amount or real time streaming data, it is not feasible to use human evaluation to separate the noisy data from the useful one because of cost and time ineffectiveness. So in order to automate the process of segregation of tweets, machine learning classification algorithm can be used to assign labels to unlabeled data with the help of trained labeled data. Supervised Machine Learning Algorithms such as Naïve Bayes, Decision Tree, K nearest neighbors, neural networks, random forest etc. generate a model on the labeled data in the training dataset and using this generated model it assigns labels to the unlabeled data in the testing dataset. In most of the cases, the training model generated is kept the same and new testing dataset is introduced for classification every time new tweets are extracted for a fixed duration of time. Such a model can be termed as” train once classify ever” where the same training model can be used to assign labels in the testing dataset. In this way, we can reduce human involvement by labeling data once and then use machine learning algorithms to assign labels to the unlabeled data.

As stated above Twitter data has evolving or changing characteristic. This characteristic of Twitter might be not suitable for train once classify ever mechanism which is commonly used in machine learning where the training dataset is kept constant and the testing dataset is varied. If we apply the same mechanism for twitter data, wherein we generate a model with tweets collected from a particular duration of time and use it to assign labels on data collected sometime far later in time, the performance of the machine learning algorithm might not be optimum. So in order to study the performance of the machine learning algorithms as the duration or time gap between the training and testing data increases we have extracted tweets containing the three hashtags namely, Red Tide, Cyanobacteria and Algae Bloom. The reason behind the decrease might be because of less similar events in the training and testing dataset. In the following sections, we have described the experimental setup for studying the characteristics of machine learning algorithms on real time and evolving Twitter data and also plotted the results of the experiments performed.



*Figure 8: Same month Experimental Setup for Machine Learning Characteristics*



*Figure 9: Different month Experimental Setup for Machine Learning Characteristics*

**6.1 Experimental Setup:**

The experiments we have implemented are based on tweets extracted every month for all three hashtags. We have considered monthly tweets for the three hashtags namely, Red Tide, Cyanobacteria and Algae Bloom. The raw tweets that are extracted need to go through a few preprocessing steps which include:

1. Removing special characters like #, @,! etc.
2. Removing Retweets or RTs
3. Converting tweets to lower case

Figure 8 and 9 show the number of relevant and irrelevant tweets used in the training and testing datasets for the hashtag Algae Bloom. As stated earlier we have assigned the labels to the tweets with the help of five human evaluators who were given detailed description for classifying the tweets. The performance of the machine learning algorithm is analyzed by gradually increasing the monthly gap between the training and testing dataset. We have pointed out the possible shortcoming of the traditional “train once classify ever” training model when used on data coming from Real time streaming source like twitter. The whole idea of the monthly analysis of machine learning algorithms is as follows:

The monthly gap between the training and testing dataset is gradually increased and the “train once classify ever” model is analyzed. Our data extraction was carried out for a duration of seven months starting from the month of September 2014 to March 2015. Our set of experiments consists of a constant training dataset obtained from the tweets of a particular month, and the testing dataset were varied by gradually increasing the monthly gap between the training and testing set, starting from 0 to 6. For instance, one of our experiments had September data as training dataset and so the testing dataset is gradually increased from September to March, where the gap increases from 0-6. There are two types of experiments:

* Same month experiments and
* Different month experiments.

Same month experiments had training and testing dataset from the same month where the monthly gap is 0. In this case, the training and testing dataset were generated by the 70-30% split mechanism where 70% data goes to the training dataset and 30% goes to the testing dataset. By 70-30% split we mean that 70% of relevant tweets go to the training set and 30% of relevant tweets go to testing dataset. Similarly for irrelevant tweets, the 70-30 division is carried out for generating the training and testing dataset. Figure 8 shows the same month experiment setup for the hashtag algae bloom.

For the different month experiments, as the name says the training and testing dataset come from different months. And here, all the tweets from the training month formed the training dataset and all the tweets from testing month formed the testing dataset. Let us consider an example from the set of experiments where September is used as training dataset and the testing dataset is gradually increased from September-March. All the subset of experiments from October to March will come under the type of Different month experiment and for September-September same month experiment mechanism is implemented. Figure 9 show the number of tweets for different month experiments for the hashtag algae bloom.

**Boosting:** The number of tweets that are posted every month depends on the occurrence of an event and its intensity. For this reason, the number of tweets extracted will vary as per the occurrence and intensity of event. In order to keep the dataset consistent, it becomes necessary to make the number of tweets, especially the number of relevant and irrelevant tweets in the training dataset same. So in order to keep the number of relevant and irrelevant tweets same in the training dataset of every month, we have used the max-count technique. For each hashtag, we determined the maximum number of relevant and irrelevant tweets from all the months. Once the max-count was computed, this count was used for boosting where the relevant and irrelevant count for each month was boosted to that max-count. For Algae Bloom, the maximum relevant count is 177 and maximum irrelevant count is 142 for the month of September. So from these max-counts, we boosted the relevant and irrelevant count for each month training dataset to 177, keeping the testing dataset unchanged. Therefore, now our training dataset in both same and different month experiments will have 177 relevant and 177 irrelevant tweets. Similarly, for hashtags Cyanobacteria and Red tide, boosting mechanism was implemented by computing the max counts. Once the dataset is ready after boosting, we have implemented two machine learning algorithms using the Weka [(Mark Hall 200](#h.2jxsxqh)[9](#_ENREF_17)) data mining tool.

The two algorithms we used for classification were Naïve Bayes and Voting. Voting algorithm had a combination of two machine learning algorithms namely Naïve Bayes and SVM (Support Vector Machine). We have considered the average of Naïve Bayes and voting algorithms for the purpose of analysis. Once the boosted dataset is prepared for each hashtag, the above mentioned machine learning algorithms are implemented on Weka, and a training model is generated on the boosted training dataset. This model is then tested on the testing dataset depending on whether it is a same month or different month experiment. The Weka tool then gives the performance results in terms of Precision, Recall, F-Measure and correctly classified instances. These measures give us the efficiency of using the same training dataset on different testing datasets. We have determined these measures for all three hashtags using both the machine learning algorithms and computed the average of these algorithms for the purpose of analysis.

**6.2 Results and evaluations**

The results obtained from the above mentioned experiments were represented in the form of a matrix. The rows of this matrix represented the months from September to March and the columns represented the difference in months of the training and testing dataset. The values in the matrix consist of the F-measure, Recall and Correctly Classified instances given by Weka. For instance, if the row value is October and the Column is 3, it means that October month is used as the training month and January, which is the third month from October is used as testing month. And the value at [October][3] will be F-measure, Recall or correctly classified instances for October-January experiment. In order to plot graphs of the above obtained measures, we have considered the average of each column of the matrix. Table 4, 5 and 6 show the correctly classified, F-measure and Recall for all the experiments. The Average row gives the average value for Naïve Bayes and voting which is used to plot the graphs.

Figure 10-12 show the Average F-measure, Correctly Classified Instances and Recall for the three hashtags vs. the difference in months. We see an almost decreasing trend in the performance of the machine learning algorithms as the difference in the months increasing

*Table 4: Correctly Classified Instances for two algorithms and average for each hashtag*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Difference in Months** | | | | | | |
| **Hashtag** | **Algorithms** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **Red Tide** | **Naïve Bayes** | 81.0 | 65.2 | 75.0 | 60.3 | 55.7 | 28.3 | 44.0 |
| **Voting** | 91.3 | 92.2 | 94.4 | 86.5 | 88.3 | 85.4 | 96.8 |
| **Average** | 86.1 | 78.7 | 84.7 | 73.4 | 72.0 | 56.9 | 70.4 |
| **Algae Bloom** | **Naïve Bayes** | 79.9 | 79.2 | 75.4 | 74.9 | 68.5 | 75.3 | 63.0 |
| **Voting** | 80.7 | 81.8 | 84.5 | 85.9 | 71.7 | 72.4 | 67.1 |
| **Average** | 80.3 | 80.5 | 80.0 | 80.4 | 70.1 | 73.9 | 65.0 |
| **Cyanobacteria** | **Naïve Bayes** | 56.5 | 57.2 | 61.8 | 57.1 | 56.5 | 79.7 | 75.2 |
| **Voting** | 67.5 | 69.4 | 72.7 | 76.3 | 75.2 | 78.4 | 80.6 |
| **Average** | 62.0 | 63.3 | 67.3 | 66.7 | 65.9 | 79.1 | 77.9 |

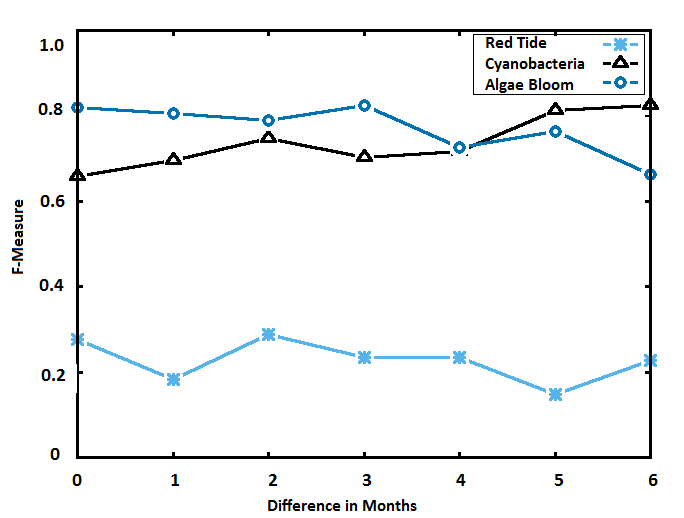
*Table 5: F-Measure for two algorithms and average for each hashtag*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Difference in Months** | | | | | | |
| **Hashtag** | **Algorithms** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **Red Tide** | **Naïve Bayes** | 0.321 | 0.208 | 0.2857 | 0.227 | 0.231 | 0.1 | 0.085 |
| **Voting** | 0.277 | 0.182 | 0.287 | 0.233 | 0.234 | 0.147 | 0.227 |
| **Average** | 0.233 | 0.156 | 0.2883 | 0.24 | 0.236 | 0.195 | 0.368 |
| **Algae Bloom** | **Naïve Bayes** | 0.832 | 0.822 | 0.7887 | 0.826 | 0.829 | 0.821 | 0.732 |
| **Voting** | 0.809 | 0.792 | 0.789 | 0.871 | 0.672 | 0.708 | 0.594 |
| **Average** | 0.821 | 0.805 | 0.7888 | 0.826 | 0.727 | 0.765 | 0.663 |
| **Cyanobacteria** | **Naïve Bayes** | 0.578 | 0.618 | 0.6767 | 0.571 | 0.617 | 0.788 | 0.845 |
| **Voting** | 0.737 | 0.773 | 0.816 | 0.833 | 0.815 | 0.837 | 0.802 |
| **Average** | 0.658 | 0.696 | 0.7463 | 0.702 | 0.716 | 0.812 | 0.824 |

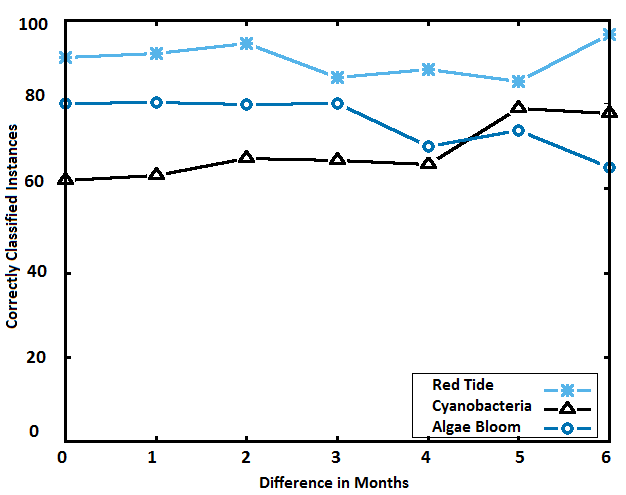
*Table 6: Recall for two algorithms and average for each hashtag*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Difference in Months** | | | | | | |
| **Hashtag** | **Algorithms** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **Red Tide** | **Naïve Bayes** | 0.594 | 0.647 | 0.6033 | 0.596 | 0.772 | 0.873 | 0.865 |
| **Voting** | 0.217 | 0.118 | 0.1953 | 0.211 | 0.177 | 0.222 | 0.311 |
| **Average** | 0.405 | 0.383 | 0.3993 | 0.403 | 0.474 | 0.548 | 0.588 |
| **Algae Bloom** | **Naïve Bayes** | 0.949 | 0.912 | 0.8757 | 0.945 | 0.963 | 0.929 | 0.965 |
| **Voting** | 0.825 | 0.727 | 0.7297 | 0.823 | 0.599 | 0.577 | 0.46 |
| **Average** | 0.887 | 0.819 | 0.8027 | 0.884 | 0.781 | 0.753 | 0.713 |
| **Cyanobacteria** | **Naïve Bayes** | 0.473 | 0.513 | 0.5657 | 0.44 | 0.538 | 0.708 | 0.767 |
| **Voting** | 0.653 | 0.708 | 0.783 | 0.883 | 0.843 | 0.77 | 0.752 |
| **Average** | 0.563 | 0.611 | 0.6743 | 0.661 | 0.691 | 0.739 | 0.76 |

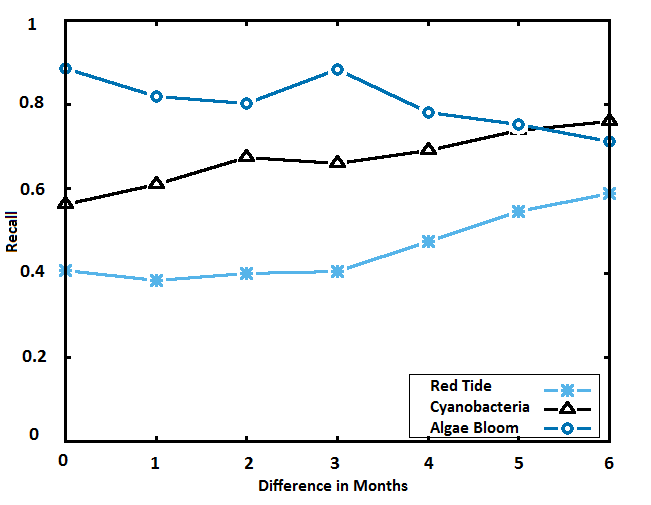
for the hashtags Red tide and algae bloom. From this we can infer that, when we use dataset from the past to classify the present data, the accuracy or performance of the machine learning is hampered. The main reason for this might be the changing characteristics of Discussion topics in the newly obtained tweets. So there might be new and latest tweets in the testing dataset, which are not part of the training model, generated which results in the incorrect classification of the new tweets. Therefore lesser the monthly gap between the training and testing dataset, more similar events will exist due to which machine learning algorithms give good classification results. But as the monthly gap increases, the similarity between the training and testing months reduces which causes decrease in the performance of the machine learning algorithms. But for the hashtag cyanobacteria, which contain less noise as it is more technical term, the machine learning algorithms have an increasing trend as the monthly gap between the training and testing dataset increases.



*Figure 10: Average F-Measure Vs. Difference in Months*



*Figure 11: Average Correctly Classified Instances Vs. Difference in Months*

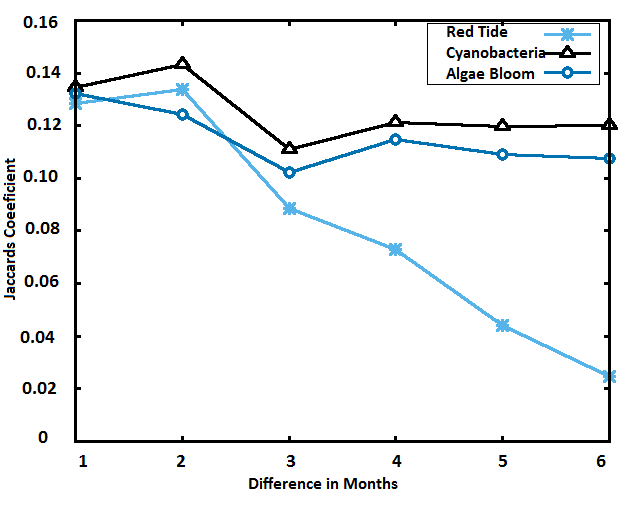


*Figure 12: Average Recall Vs. Difference in Months*

Weka data mining tool for machine learning gives the F-measure Recall and Precision for both the labels relevant and irrelevant individually. We have plotted the graph for only relevant Recall and F-Measure. Correctly classified instances gives the measure of how many tweets were correctly labeled.

In order to find the reason behind the decrease in performance of machine learning algorithms, we have performed another set of experiments where, we found the overlapping words occurring in the tweets from different months that were used as the training and testing months in the previous machine learning experiments.

This will give the similarity of tweets between two months. For each month, we determined unique words from all the tweets occurring in a month after removing the stop words and determined the overlapping words between the two months.



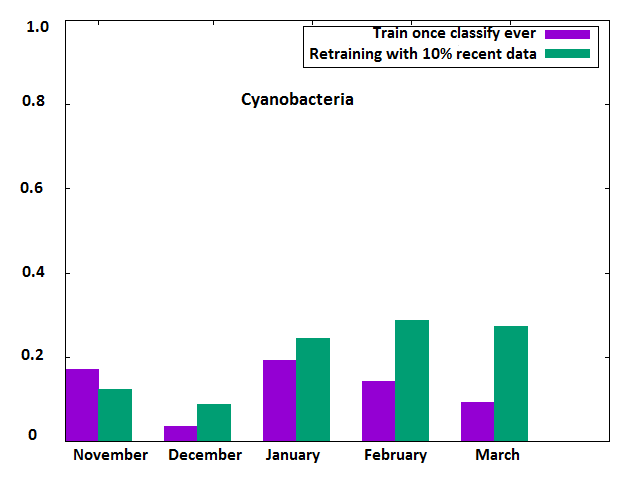
*Figure 13: Jaccard’s Coefficient Vs. Difference in Months*

We used the same combination of training and testing dataset months as used in the machine learning experiments and also used the same format of matrix with the rows as months and columns from 1-6 which denote the gap in the training and testing months. This matrix is then filled with the Jaccard’s coefficient, which is determined as follows:

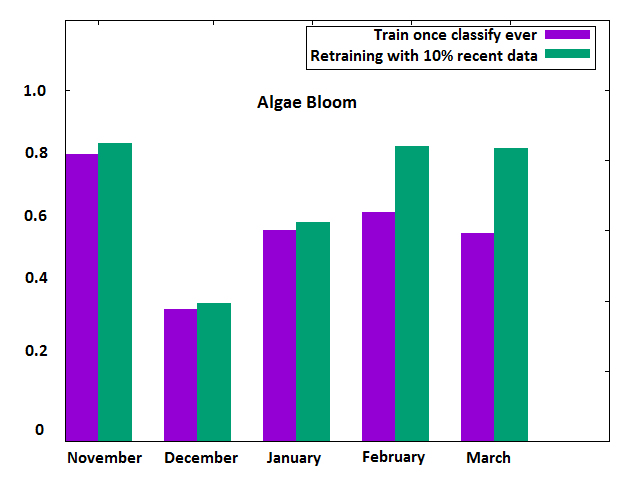
where, M1 and M2 are months whose overlapping words are determined. M1∩M2 gives the overlap of words between the two months M1 and M2, and M1 U M2 is the sum of unique words in M1 and M2. From Figure 13, we can see that as the gap between the months increases, the overlap count or Jaccard’s coefficient decreases. This shows that there might be a decrease in similarity of events or tweets as the gap increases which might be one of the reason behind the decrease in performance of the machine learning algorithms.

**6.3 Retraining:**

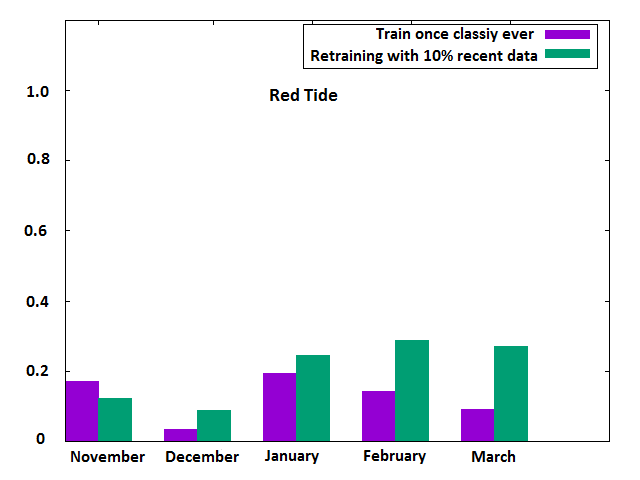
As seen in the above machine learning section, the performance of the machine learning algorithms show an almost decreasing trend as the monthly gap between the training and testing dataset increases. This shows that the train once classify ever model is not suitable for the data from social networking sites like Twitter where the data is evolving and has changing characteristics. In order to improve the performance of the machine learning algorithms, we need to retrain the training model when new data from the social media is collected. In this way, the training dataset will not only have data from past but will also have recent data. Such retrained training model will result in more similarity between the training and testing dataset and will help in improving the performance of the machine learning algorithms. But as we know for supervised training model, we need to use a training dataset that already has labels assigned to it. For large amount of real time tweets it is not feasible to label tweets as and when they are extracted. And the whole reason behind using machine learning for classification was to reduce human involvement in labeling the relevant and irrelevant tweets. But for the retraining model, we will need more human involvement for labeling, every time new tweets are extracted which is not feasible for big data.



*Figure 14: Precision improvement after using retraining model*



*Figure 15: Precision improvement after using retraining model*



*Figure 16: Precision improvement after using retraining model*

One of the possible way out would be to use only a fraction of tweets from a month for training. In this way, we will not only get latest tweets in the retraining model but also will

reduce human involvement to label only a fraction of the new incoming tweets rather than labeling all newly extracted tweets.

Figure 14-16 show the increase in performance of machine learning algorithms when the above mentioned retraining model was used. In order to compare the performance of both the training models, we have plotted bar graphs that clearly show the performance increase of the retraining model. The horizontal axis gives the months that were used for testing. For each testing month in the graph, the left bar denotes precision when September was used for training and the right bar denotes retraining model where the training dataset consist of all months from September to a month prior to the testing month.

For instance, when testing month is January, the retraining model will consist of fraction of tweets from each individual month in September-December, whereas the train once classify ever model will consist of tweets only from September. For the retraining model, each month used in the training dataset will consist of only a fraction of tweets for training. For uniform comparison between the two training models, the number of tweets in the training dataset of both models is the same. From the graphs plotted in Figures 12-14 we can clearly see an increase in the performance of the machine learning algorithms when the retrained model is used in the training dataset. This shows that for evolving data source like Twitter where the data contents keep varying, machine learning performance can be improved by using such a retraining model and at the same time reduce human involvement by labeling only a fraction of tweets from each month.

CHAPTER 7

CONCLUSION

We have performed a detailed analysis of how social media is used for the environmental monitoring of CyanoHABs and have also put forth how the use of hashtag plays an important role in this monitoring. We have used four different hashtags that are associated with the same phenomenon of CyanoHABs growing on water bodies namely, Cyanobacteria, Blue Green Algae, Red Tide and Algae Bloom. We studied how technicality of the hashtag used has an impact on the number of tweets obtained, number of users and the professional background of the users. More technical term like Cyanobacteria, will have less number of tweets, less number of contributors who are experts or domain scientists but the information obtained will have high trustworthiness and reliability. Whereas for generic terms like Red Tide, the number of tweets will be more, the number of contributors will be more but the information obtained will have less reliability because of the majority of noisy information and contributors being general public.

We also analyzed how machine learning algorithms show an almost decreasing performance trend when the monthly gap between the training and testing dataset increases. We have used the tweets based on the slowly evolving CyanoHABs on water bodies and showed that the “train once classify ever” model of machine learning has flux when it comes to data from social media which has real time and changing characteristics.

We have also proposed a retraining model that will help overcome the shortcomings of the traditional train once classify always model, where we consider only a fraction of tweets from all months for training. This will not only result in better performance of machine learning algorithm with recent data in training model, but will also reduce the human involvement required in supervised machine learning algorithms.

REFERENCES

1. <http://www.webmd.com/mental-health/>
2. Sayali Kale UGA Graduate Student Computer Science.
3. ‘Using Twitter to Track the Flu (“Dredze et al 2014”)

# Analysis of Twitter posts could provide fresh insight into mental illness trends (“Dredze et al 2015”)

1. Discovering Multi-Scale Co-occurrence Patterns of Asthma and Influenza with the Oak Ridge Bio-surveillance Toolkit." *(“Ramanathan et al* 2015”)
2. ‘SemEval-2015 Sentiment analysis in twitter (“*Rosentha et al 2015”* )
3. “Retweets but not just Retweets - Quantifying and predicting influence on twitter (Rosemann et al., 2012).

# Detecting suicidality on Twitter (“O'Dea et al 2015”

1. Mining Twitter Data to Improve Detection of Schizophrenia (“McManus et al-2015”)
2. Measuring Post Traumatic Stress Disorder in Twitter (“Coppersmith et al - 2014”)
3. <http://spark.apache.org/>
4. <https://radimrehurek.com/gensim/>
5. <https://about.twitter.com/company>
6. <http://hadoop.apache.org/>
7. Predicting Postpartum Changes in emotion and behavior via Social Media (“Choudhury et al-2013”)
8. Apoorv Agarwal, Fadi Biadsy, and Kathleen Mckeown. 2009. Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams. Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009), pages 24–32, March.
9. Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pages 36–44.
10. Adam Bermingham and Alan Smeaton. 2010. Classifying sentiment in microblogs: is brevity an advantage is brevity an advantage? ACM, pages 1833–1836.
11. C. Fellbaum. 1998. Wordnet, an electronic lexical database. MIT Press.
12. Michael Gamon. 2004. Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. Proceedings of the 20th intern
13. Munmun De Choudhury, Scott Counts, and Eric Horvitz. 2013a. Major life changes and behavioral markers in social media: Case of childbirth. In Proceedings of the ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW).
14. Munmun De Choudhury, Scott Counts, and Eric Horvitz. 2013b. Predicting postpartum changes in emotion and behavior via social media. In Proceedings of the ACM Annual Conference on Human Factors in Computing Systems (CHI), pages 3267–3276. ACM.
15. Munmun De Choudhury, Scott Counts, and Eric Horvitz. 2013c. Social media as a measurement tool of depression in populations. In Proceedings of the Annual ACM Web Science Conference.
16. Munmun De Choudhury, Michael Gamon, Scott Counts, and Eric Horvitz. 2013d. Predicting depression via social media. In Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM).
17. Munmun De Choudhury, Andres Monroy-Hernandez, and Gloria Mark. 2014. ” narco” emotions: Affect and desensitization in social media during the mexican drug war. Munmun De Choudhury. 2013. Role of social media in tackling challenges in mental health. In Proceedings of the 2nd International Workshop on SociallyAware Multimedia, pages 49–52.
18. TextBlob: <http://textblob.readthedocs.org/en/dev/quickstart.html#sentiment-analysis>
19. <http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010>
20. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. The Journal of Machine Learning Research, 3:993–1022, 2003
21. Liangjie Hong and Brian D. Davison. Empirical study of topic modelling in twitter
22. K-Mean: <https://en.wikipedia.org/wiki/K-means_clustering>
23. <https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm>
24. <http://scikit-learn.org/stable/modules/classes.html#module-sklearn.cluster>
25. <https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding>
26. <https://www.digitalocean.com/>
27. <https://aws.amazon.com/ec2/>
28. <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html>
29. <https://gephi.org/>