Predicting Sentiment in Social Media

By

Pavan Nutalapati

Advisor

Dr. Christopher Homan

Department of Computer Science

B. Thomas Golisano College of Computing and Information Sciences

Rochester Institute of Technology

Rochester, New York

December 2013

Contents

Acknowledgement

I would like to sincerely thank my project guide Dr Christopher Homan for effectively guiding me throughout this project, without his enthusiasm and timely support it would have been impossible to successfully complete this project. Also special thanks to Dr Joe Gigel for his invaluable suggestion during the project milestone discussions.

1. Introduction

Emotional health is an important aspect for the well being of individuals. Various studies shows that the millions of people suffer from emotional disorders [1] and only little percentage of them are identified and diagnosed effectively. The main hurdle of identifying and analyzing these phenomena is because of paucity of data. Due to the social stigma and other un-awareness reasons [2], mental health issues are less reported.

Current methodologies of gathering emotional information about the individuals are mostly based on surveys and information sourced from health care providers; such approaches are error prone and not scalable, they also limit the analysis on distinguishing the cause from effect [2]. Moreover the data provided by this traditional approach would be in the aggregated from which limits the scope for distinguishing the cause from effect, e.g. predicting the sentiment of individuals based on their social interactions, mobility, Etcetera.

As social media usage is rapidly growing globally, millions of users have been sharing their thoughts in the form of message postings. One such example is Twitter, a microbloging website used by 500 million subscribers churning out 360 millions messages a day. Some studies [3] have shown that this emotionally rich can be used effectively to overcome the barrier of shortage of data.

In this project we have worked on developing a model to quantify and predict the sentiment of individuals without direction intervention using the data from Twitter social media. The work particularly focuses on developing a statistical classifier tailored for twitter messages, which can infer the sentiment value of each tweet (message) based on its content and classify them into either of two categories namely positive sentiment and negative sentiment.

This work is an important step to understand the emotion state of each user and to predict his future state by analyzing his social structure. We have compared our results with the LIWC classifier, which allocates the sentiment score for each message based on the presence of certain keywords present in the LIWC sentiment dictionary [6]. The experiments performed on the developed classifier showed a precision value of 70% and a correlation of around 0.4 with LIWC calculated values.

The paper is arranged in the following manner: Section 2 describes the existing techniques used for sentiment analysis. Section 3 explains in detail about implementation part of the project. Section 4 explains about the results of the experiments and Section 5 talks about the conclusion.

2. Background

With the advent of rich data set from social media, several research works has been conducted with focus on harnessing them to predict a variety of phenomenon in the domains of sentiment analysis, public health, elections and many more [5]. This section discusses some of the important work among them which are close to this project.

In the paper by Karla [4], the authors have collected the twitter data of users around the New York City and developed a model to generate a sentiment map which depicts the sentiment of users on fine grained spatial and temporal scales. In this experiment, the authors have collected approximately 600K tweets using the twitter API which have the geo coordinates tagged to it. The geo tags were required to do the analysis on the spatial scale. Upon sourcing the data they have developed a statistical classifier to infer the sentiment of each tweet based on its content.

In machine learning context, classification is a problem to identify the category of the new observations (test data) based on the learning from the training set whose category information is already known. Statistical classifier is a mathematical model which performs this task of classification. As shown in the below figure, the training data set is fed to classifier which learns from the pattern and prepares a model.

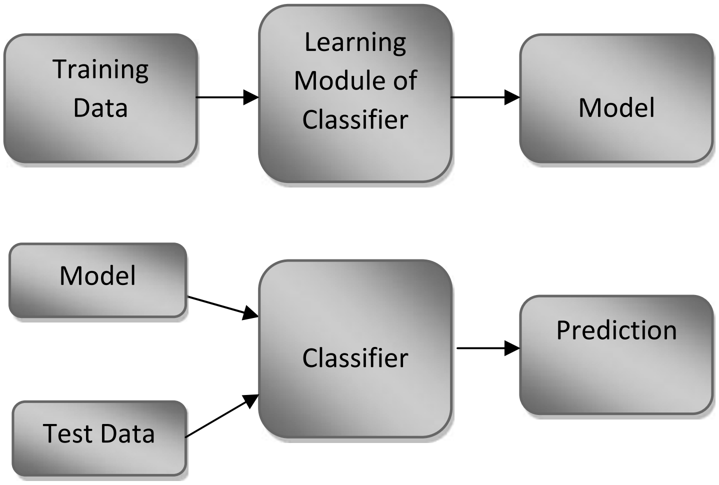
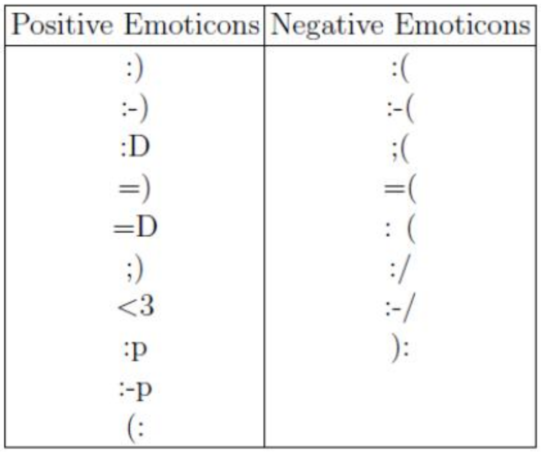


Figure 1: Supervised learning Classifier

The classifier developed by Karla [4] was based on supervised learning, where the training data set was formed from the tweets containing the emoticons.



Positive Emoticons	Negative Emoticons
:)	:(
:-)	:-(
:D	;(
=)	=(
=D	: (
;)	:/
<3	:-/
:p	):
:-p	
(:	

Figure 2: Emoticons used for the creation of training set by Karla [4]

The text content of each tweet was parsed to check for the presence of emoticons shown in the figure 2. Training set containing such emoticons was labeled into two categories namely positive sentiment and negative sentiment depending upon the group to which the emoticon belongs to. To classify the unlabeled tweets, first there were standardized by removing special characters and then tokenized them using Natural language toolkit to form the features. This feature data set was fed into two Bayes classifiers which labeled the tweet with either positive or negative sentiments.

After identifying the sentiment of each tweet, they used the geo tag information included in the tweet to do analysis on the spatial and temporal scales. They have found interesting pattern where the positive sentiment was higher in the land mark areas such as Times Square and public parks, and strong negative sentiment in the areas closer to cemeteries, hospitals and jails. On the temporal scale, the tweets with positive mood were generally found more in the weekends and during the midnight.

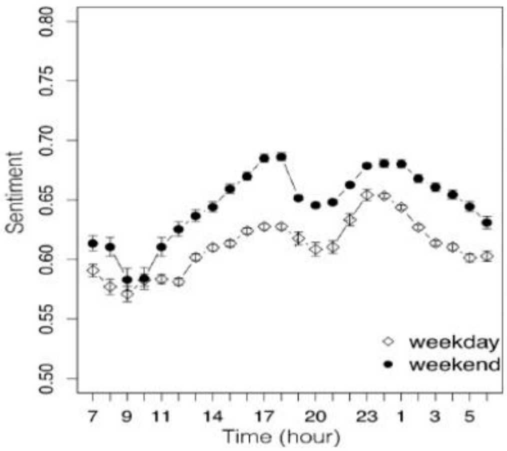


Figure 2: Example of emotion map based on time by Karla [4]

We have incorporated the idea of creation of training set based on the presence of emoticons as described by the paper by Karla[4].

Sadliek and collaborators [5] used the twitter data to model the spread of diseases based on the social interactions. The data set consists of 2.5 million geo tagged tweets originated from the New York City. Their model successfully analyzed how diseases are transmitted among the users based on their social interactions and the geographic collocations. Their work is based on the bottom up approach similar to the approach take in our project, where the sick users were identified based on the tweet postings and then used it along with the geo tag data to analyze the transmission of disease among the users. The authors claim that this is a novel approach in the study of disease vectors.

To identify the sickness message they used cascading SVM (support vector machine)[7] based model for the classification, the reason specified for the choice of SVM algorithm is that the it shows good performance with high dimensional data and efficiently solves the class imbalance problem, meaning that the tweets of interest(health related) are very less compared to the normal types.

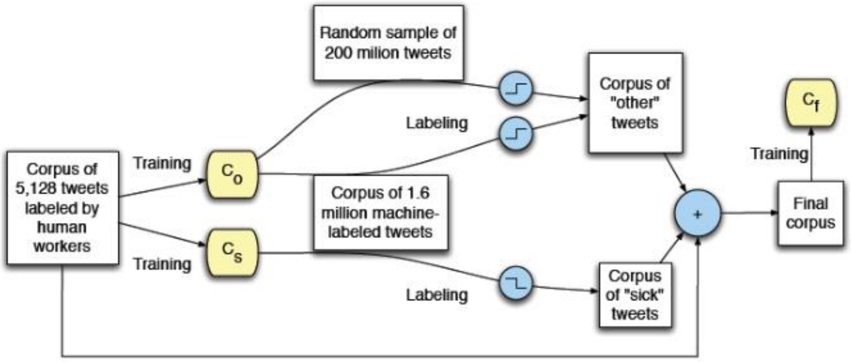

Random sample of 200 milion tweets
Corpus of "other" tweets
Labeling
C_f
Training
Corpus of 5,128 tweets labeled by human workers
Training
C_o
Corpus of 1.6 million machine-labeled tweets
+
Final corpus
C_s
Training
Labeling
Corpus of "sick" tweets

Figure 3: Semi Supervised based cascading SVM classifier used by Sadliek [5]

To create training set, they used bootstrapping method, where they have created two classifiers shown as C_o and C_s in the figure 3, these supporting classifiers are being used for the creation of training set with high confidence values for the main classifier $C_{f.}$ The initial training data for the supporting classifiers have been sourced from Amazon Mechanical Turk workers [8], a crowd sourcing market place with a web services interface. By this process they have procured 5,128 tweets with high precision information about the type. For building the classifier $\text{SVM}^{\text{light}}$ package has been used, it is an open source implementation of Support vector machine (SVM).

SVM algorithm requires the data to be presented in the form of feature sets; the authors have normalized the content of tweets by removing special characters and URLs as a first step, then the text was tokenized to form features containing the tuples of unigrams, bigrams and trigrams.

For example, for the tweet “I am happy”, the features set are:

{“I”, “am”, “happy”, “I am”, “am happy”, “I am happy”}

Because of this technique, the feature set is of high dimensions, where each feature can be imagined as dimension in the Cartesian space and the tweet as a point in the space. To deal with the high dimension they have optimized rocarea measure of SVM.

Using this C_f classifier, the sick users are identified based on the classification of the tweets they posted. As they had the data about the social network of each user and the location tags of GPS, they have created a model correlating this information. The authors were successfully able to establish a direct relationship between the interactions among the users and the eventual disease transmission within them.

In another important work by de choudary and others [1], they have created a model to identify the emotional disorders among the people and the factors influencing it. The classifier was based on SVM model which have four classes namely positive effect, negative effect, activation and dominance. They also have used LIWC, linguistic analysis software for the classification. The classification had a higher accuracy of 70%.

In a similar research work on the sentiment analysis by Christopher Homan and others[2], the authors have investigated the use of facebook (another social media) data to quantify and predict the mental state of individuals on a temporal scale, i.e. the mood dynamics on a time scale like hourly, daily and weekly basis. The model was based on the concept of principal component analysis and Fourier analysis. The model was able to predict the mood of individuals with an accuracy of 61%. Further their experiments were able to analyze the relationship between the sentiment of users based on their social structure and the timing of their postings. One important difference between the work by Homan[2] and the de choudary is that the later didn’t validated the data using the Mechanical Turks, as they claim that the data can be biased if the workers themselves suffer from depression.

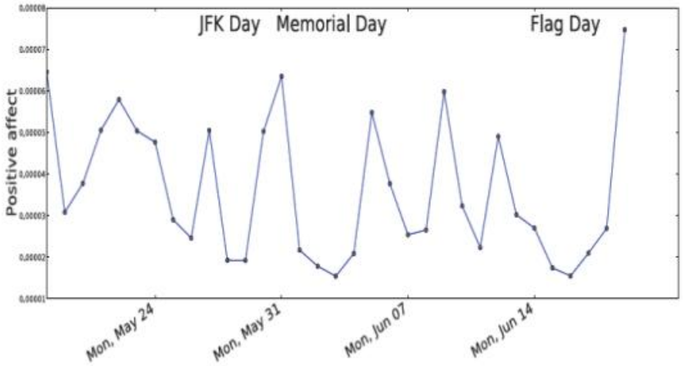


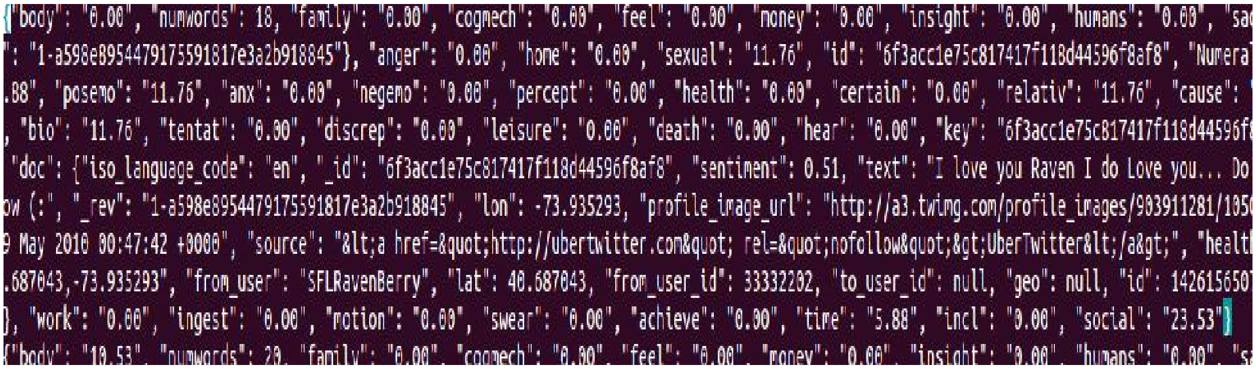
Figure 4: Example of sentiment analysis on a temporal scale by Homan[2]

To conclude this section, this project has taken the cues from the works done by Sadliek [5], Homan [2] and Karla [4].where we have created a classification model to predict the sentiment of users based on the content of the tweet. We have used SVM^{light} tool for building the classifier. The training set was formed by segregating the tweets containing the emoticons shown in the figure [2]. This work is an important step towards the development of model to analyze the relationships between the emotional state of the users and their social structures.

3. Data and Tools

3.1 Data Sets

The experiment is based on the data from Twitter micro blogging website. Data set consist of 2,361,679 tweets arranged in JSON format as shown below.



```
{"body": "0.00", "numwords": 18, "family": "0.00", "cogmech": "0.00", "feel": "0.00", "money": "0.00", "insight": "0.00", "humans": "0.00", "sad
": "1-a598e8954479175591817e3a2b918845"}, "anger": "0.00", "home": "0.00", "sexual": "11.76", "id": "6f3acc1e75c817417f118d44596f8af8", "Numera
.88", "posemo": "11.76", "anx": "0.00", "negemo": "0.00", "percept": "0.00", "health": "0.00", "certain": "0.00", "relativ": "11.76", "cause": "
, "bio": "11.76", "tentat": "0.00", "discrep": "0.00", "leisure": "0.00", "death": "0.00", "hear": "0.00", "key": "6f3acc1e75c817417f118d44596f
 "doc": {"iso_language_code": "en", "_id": "6f3acc1e75c817417f118d44596f8af8", "sentiment": 0.51, "text": "I love you Raven I do Love you... Do
ow (:", "_rev": "1-a598e8954479175591817e3a2b918845", "lon": -73.935293, "profile_image_url": "http://a3.twimg.com/profile_images/903911281/1050
9 May 2010 00:47:42 +0000", "source": "&lt;a href=&quot;http://ubertwitter.com&quot; rel=&quot;nofollow&quot;&gt;UberTwitter&lt;/a&gt;", "healt
.687043,-73.935293", "from_user": "SFLRavenBerry", "lat": 40.687043, "from_user_id": 33332202, "to_user_id": null, "geo": null, "id": 142615650
}, "work": "0.00", "ingest": "0.00", "motion": "0.00", "swear": "0.00", "achieve": "0.00", "time": "5.88", "incl": "0.00", "social": "23.53"}
{"body": "10.53", "numwords": 20, "family": "0.00", "cogmech": "0.00", "feel": "0.00", "money": "0.00", "insight": "0.00", "humans": "0.00", "sa
```

Figure 5: Snapshot of Twitter Data in JSON format

JSON is a format where the data is stored in the attribute-value pairs, as shown in the figure 5, there are numerous attributes for each tweet. The attributes which are of importance to us are,

- “Text”: This contains the actual text posted by the user
- “from_user” : User who has posted this tweet
- “posemo”:The percentage count of the text tokens matching in the posemo section of the LIWC dictionary, the score represents the intensity of positive emotion of the tweet, refer section for more details
- “negemo”:The percentage count of the text tokens matching in the negamo section of the LIWC dictionary, it represents the intensity of negative emotion of the tweet.
- “"lat": specifies the geo location of the user

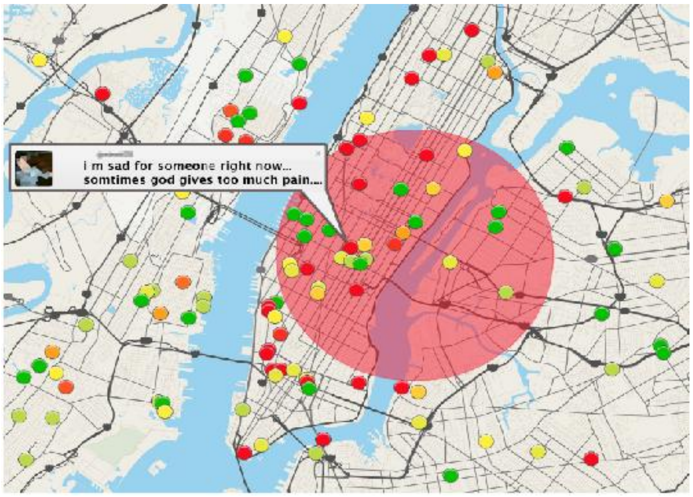
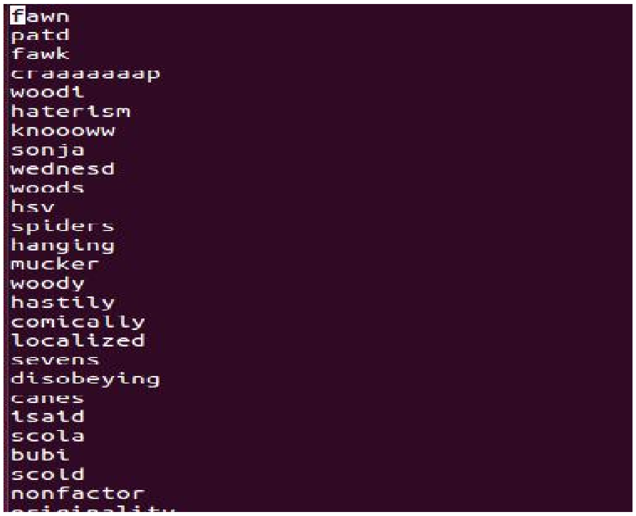

i m sad for someone right now...
somtimes god gives too much pain....

Figure 5: graphical representation of geo tagged Twitter data
(figure taken from paper by Homan [2])

Apart from the twitter data, the project needs a sentiment dictionary which contains the possible lexicons with significant sentiment value for the classification process. Figure 6 displays the sample snapshot of the data, File is named as "WORDS_all-sick.txt". there are 1742172 number of strings in the dictionary.

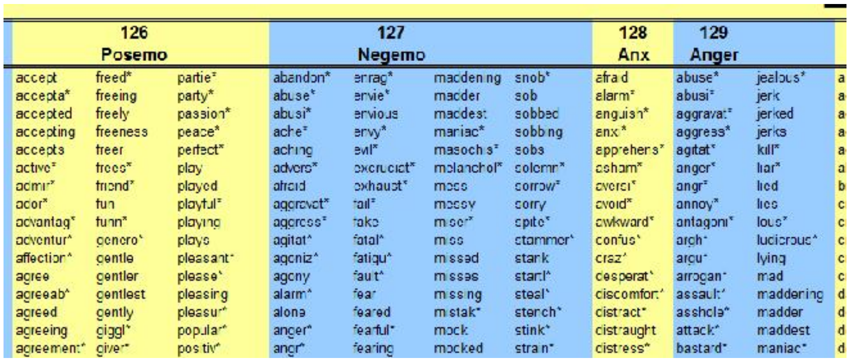


```
fawn
patd
fawk
craaaaaaap
woodi
haterism
knoooww
sonja
wednesd
woods
hsv
spiders
hanging
mucker
woody
hastily
comically
localized
sevens
disobeying
canes
isaid
scola
bubi
scold
nonfactor
originality
```

Figure 6: Sentiment Dictionary

3.2 LIWC Dictionary

LIWC , Linguistic Inquiry and Word Count is a collection of lexicons, which are categorized according to the emotional significance. As discussed in the section 3.1 this is used to calculate the posemo and negemo score of the tweets.



126			127				128	129		
Posemo			Negemo				Anx	Anger		
accept	freed*	partie*	abandon*	enrag*	maddening	snob*	afraid	abuse*	jealous*	a
accepta*	freeing	party*	abuse*	envie*	madder	sob	alarm*	abusi*	jerk	a
accepted	freely	passion*	abusi*	envious	maddest	sobbed	anguish*	aggravat*	jerked	a
accepting	freeness	peace*	ache*	envy*	maniac*	sobbing	anx*	aggress*	jerks	a
accepts	freer	perfect*	aching	evil*	masochis*	sobs	apprehens*	agitat*	kill*	a
active*	frees*	play	advers*	excruciat*	melanchol*	solemn*	asham*	anger*	liar*	a
admir*	friend*	played	afraid	exhaust*	mess	sorrow*	aversi*	angr*	lied	b
ador*	fun	playful*	aggravat*	fail*	messy	sorry	avoid*	annoy*	lies	c
advantag*	funn*	playing	aggress*	fake	miser*	spite*	awkward*	antagoni*	lous*	c
adventur*	genero*	plays	agitat*	fatal*	miss	stammer*	confus*	argh*	ludicrous*	c
affection*	gentle	pleasant*	agoniz*	fatigu*	missed	stank	craz*	argu*	lying	c
agree	gentler	please*	agony	fault*	misses	startl*	desperat*	arrogan*	mad	c
agreeab*	gentlest	pleasing	alarm*	fear	missing	steal*	discomfort*	assault*	maddening	d
agreed	gently	pleasur*	alone	feared	mistak*	stench*	distract*	asshole*	madder	d
agreeing	giggl*	popular*	anger*	fearful*	mock	stink*	distraught	attack*	maddest	d
agreement*	giver*	positiv*	angr*	fearing	mocked	strain*	distress*	bastard*	maniac*	d

Figure 6: LIWC Dictionary

3.3 Software and Tools

The project is developed in python 2.7 on Ubuntu platform, simplejson and numpy libraries are additionally installed.

SVM^{light} tool is used as a classifier, it is an open source implementation of support vector machines algorithm. It has rich collection of options which solves the classification and regression problems. The software has two modules namely svm_learn and svm_classify, the former is used for learning the labeled data to produce a model and the later is used for the labeling the test data using the already generate model. Below are the two basic commands for learning and classifying.

```
svm_learn  training_data  model_file

svm_classify  testing_data model_file predictions
```

There are two options of learning, classification and regression, in classification the prediction would be in a Boolean format and in the regression, the model assigns weights in addition to the label specification.

R is used for statistical analysis. We have performed some experiments to compare with the sentiment values already generated in the form of posemo and negemo.

The configuration and running commands of these tools which are specific to project can be found:“https://github.com/pavan449/pxn5254_F2013_project/blob/master/regression/R_SVM_Commands.txt “.

4 Methodologies

4.1 System overview

The implementation consists of various phases. Twitter data is processed to form training and test feature vectors, which are eventually fed to SVM^{light} modules to predict the sentiment value of each tweet. Figure 6 details the high level implementation model

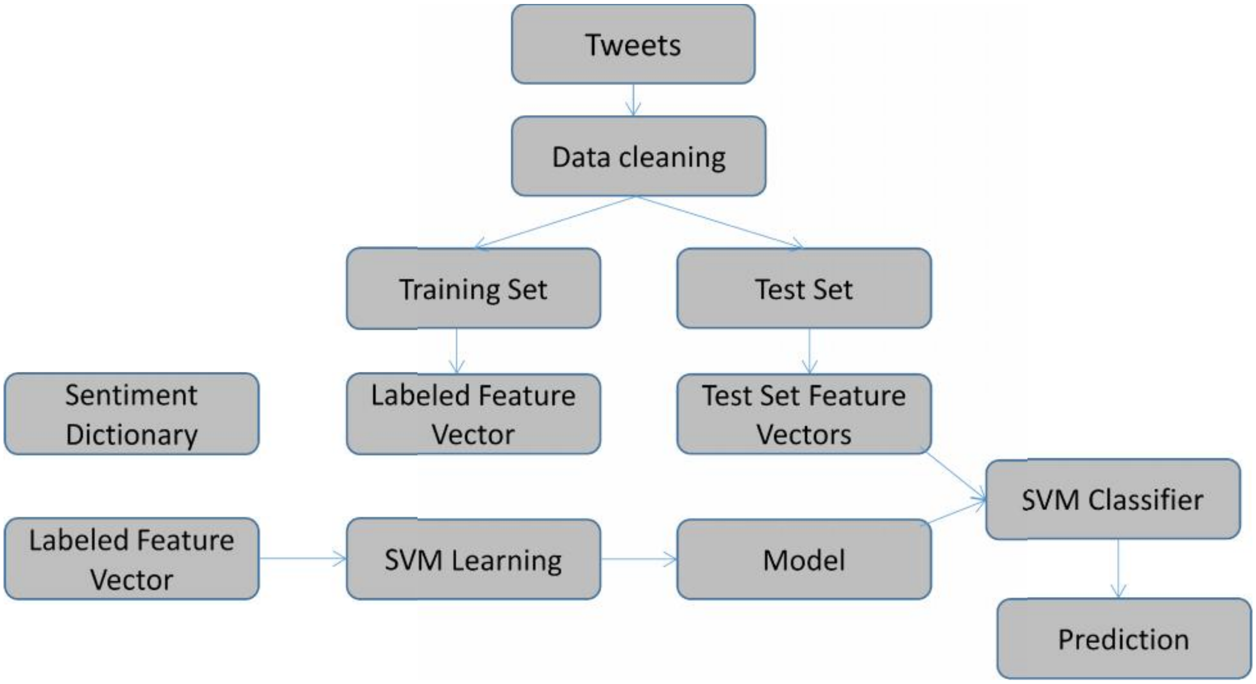


Figure 7: High Level System Architecture

4.2 Implementation

In the initial step, Twitter data stored in JSON format is parsed to check for the presence of emoticons listed in the figure 2. Tweets containing the emoticons are moved into a different file which forms the training set for our classifier. The high level steps followed by algorithm to segregate the data are shown below.

1. Load the positive and negative emoticons list in two separate lists
2. Create three files for writing, two for storing the training tweets and the other for test data.
3. For each Tweet in the file:
    a. Load it into dictionary to query the key value pairs

b. Check for the presence of emoticons given in the lists
c. If emoticon found from the positive list write it to the file storing positive training data set
d. Else If emoticon found from the negative list write it to the file storing positive training data set
e. Else write the tweet to the file storing test data set

Once we have data segregated into training and test data sets, we need to create features sets out of them to write to the SVM classifier. As a first step, we normalize the text of tweet by removing punctuations and URLS, and then the text is tokenized to from features containing tuples of unigrams, bigrams and trigrams. This work follows a similar approach described in the paper by Sadilek[5]. The features sets of each tweet begin with a label indicating the class to which the tweet belongs to, then the tuples are indexed to the sentiment dictionary which we have discussed in the section 3. The high level pseudo code is described here:

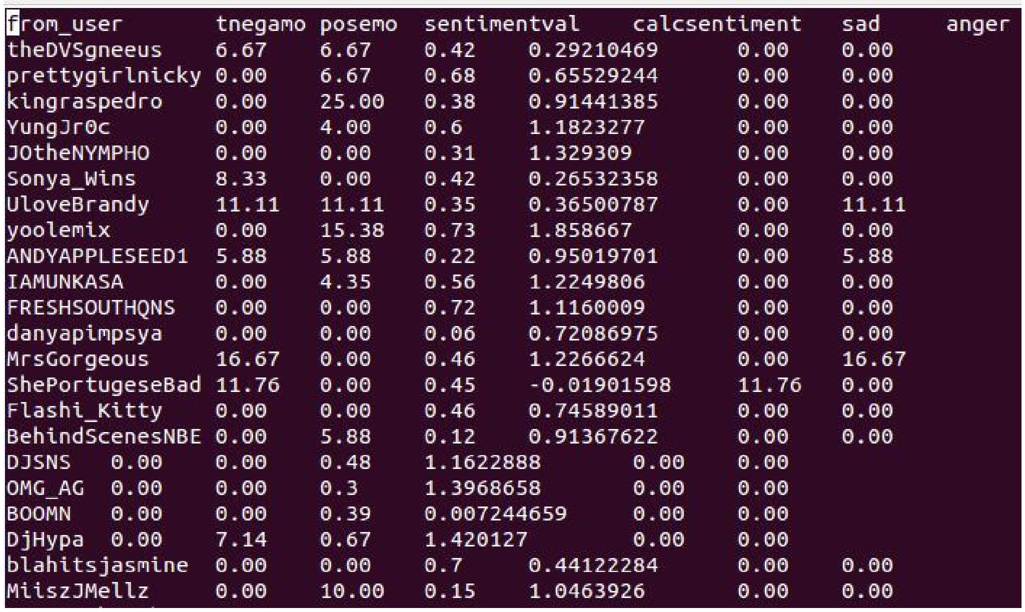
1. Load all the strings in sentiment dictionary into set data structure
2. Parse the tweets from the training and test data files and store them in dictionary.
3. Clean the text of each tweet by removing the punctuations and special characters
4. Tokenize the text of tweet into unigrams, bigrams and trigrams and store in set.
5. Take the intersection between the sentiment set and the tweet tokens and get the index of the common tokens.
6. The positive tweet is labeled with +1, negative labeled with -1 and 0 for the training data.
7. Each feature is appended with a value equal to inverse of square root of the count.

E.g. of features:

Positive feature: -1 6084:0.447214 20407:0.447214 26560:0.447214 …
Negative feature: 1 40147:0.577350 42537:0.577350 259225:0.577350…
Unlabeled feature: 0 355:0.213201 10801:0.213201 12943:0.213201…

With this we have the data in the form of features sets which can be given as input to the SVM^{light} classifier. The training data is processed by svm_learn to produce a model and the svm_classify is used to label the training data, which is effectively the sentiment value of that particular tweet, Note that by default the svm_classify works in the classifier mode, use with option “–z r” to enable in the regression mode. Once we have the prediction file generated, assign each tweet with the sentiment values from it.

To perform statistical analysis, a tab separated file containing the information of posemo, negamo, calculated sentiment value, User, Sad information of each tweet is created. Snap shot of such file is shown in the figure:



```
from_user       tnegamo posemo  sentimentval    calcsentiment   sad     anger
theDVSgneeus    6.67    6.67    0.42    0.29210469      0.00    0.00
prettygirlnicky 0.00    6.67    0.68    0.65529244      0.00    0.00
kingraspedro    0.00    25.00   0.38    0.91441385      0.00    0.00
YungJr0c        0.00    4.00    0.6     1.1823277       0.00    0.00
JOtheNYMPHO     0.00    0.00    0.31    1.329309        0.00    0.00
Sonya_Wins      8.33    0.00    0.42    0.26532358      0.00    0.00
UloveBrandy     11.11   11.11   0.35    0.36500787      0.00    11.11
yoolemix        0.00    15.38   0.73    1.858667        0.00    0.00
ANDYAPPLESEED1  5.88    5.88    0.22    0.95019701      0.00    5.88
IAMUNKASA       0.00    4.35    0.56    1.2249806       0.00    0.00
FRESHSOUTHQNS   0.00    0.00    0.72    1.1160009       0.00    0.00
danyapimpsya    0.00    0.00    0.06    0.72086975      0.00    0.00
MrsGorgeous     16.67   0.00    0.46    1.2266624       0.00    16.67
ShePortugeseBad 11.76   0.00    0.45    -0.01901598     11.76   0.00
Flashi_Kitty    0.00    0.00    0.46    0.74589011      0.00    0.00
BehindScenesNBE 0.00    5.88    0.12    0.91367622      0.00    0.00
DJSNS   0.00    0.00    0.48    1.1622888       0.00    0.00
OMG_AG  0.00    0.00    0.3     1.3968658       0.00    0.00
BOOMN   0.00    0.00    0.39    0.007244659     0.00    0.00
DjHypa  0.00    7.14    0.67    1.420127        0.00    0.00
blahitsjasmine  0.00    0.00    0.7     0.44122284      0.00    0.00
MiiszJMellz     0.00    10.00   0.15    1.0463926       0.00    0.00
```

Figure 7: snapshot of tab separate file prepared for statistical analysis

5 Results

Multiple experiments have been performed to check the performance of the developed model. The results were compared with the sentiment values generated by the LIWC classifier using the statistical analyzing tool R.

In the first test, the labeled tweets were divided into two halves to form training and test sets. The training data set was fed to SVM learning module to generate a model and then test the precision using the SVM classifier model. The results showed that the precision of the classifier is approximately 70.03 % and recall of 68.04 %. Precision and Recall are defined below. Here TP means True positive, FP is false positive and FN means false negative.

$$\text{Precision} = TP/\ (TP + FP)$$

$$\text{Recall} = TP/(TP+FN)$$

The SVM^{light} has been operated upon both in the regression and classification mode, in the regression mode the values greater than 0 are classified as positive and less than zero as negative.

In the next set of experiments, We tried to find correlation between the calculated sentiment values and LIWC generated values for posemo, negamo, sad and anger values. The experiments are performed in R.

To see the distribution of sentiment values, histogram plots have been generated as shown in the figure 8 and figure 9

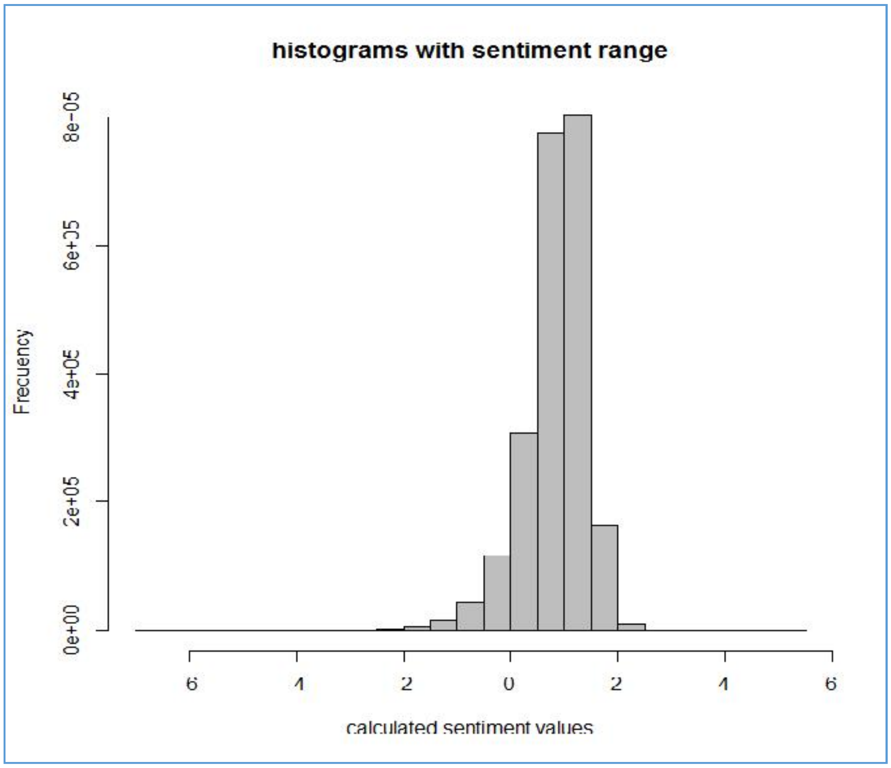


Figure 8 : Distribution of LIWC calculated sentiment values

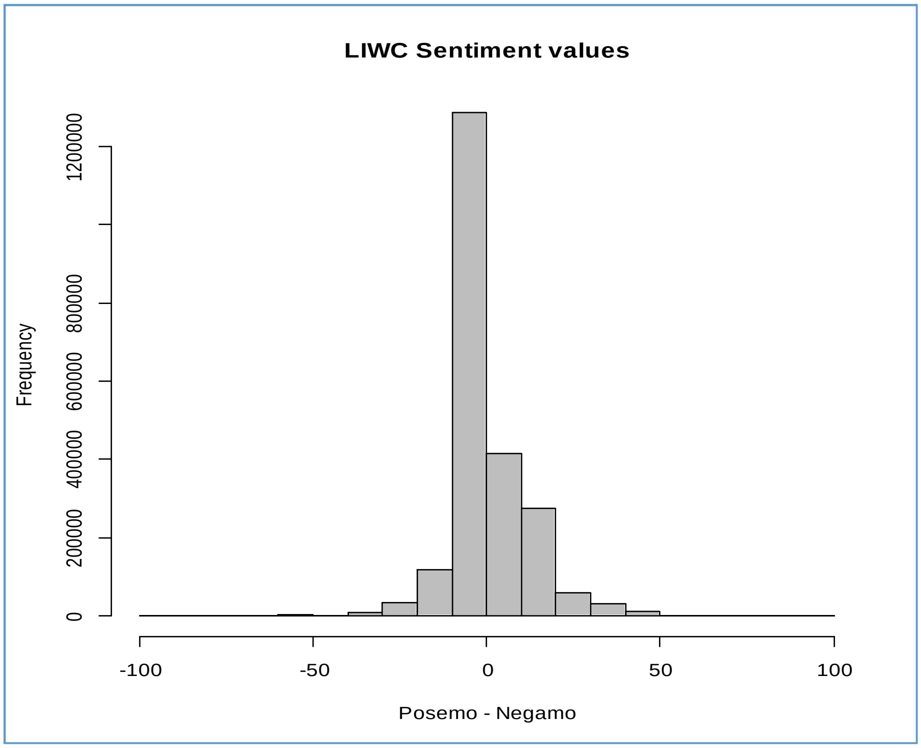


Figure 9 : Distribution of calculated sentiment values

The measure the relationship between these sentiment values and posemo, negamo correlation coefficient was calculated and we found that the value is around 0.39. And the correlation with the values sad and anger data was very small value 0.03

As the values of calculated sentiment values are continuous, direct scatter plot was too gibberish to comprehend. We have rounded up the float values to the nearest integer value.

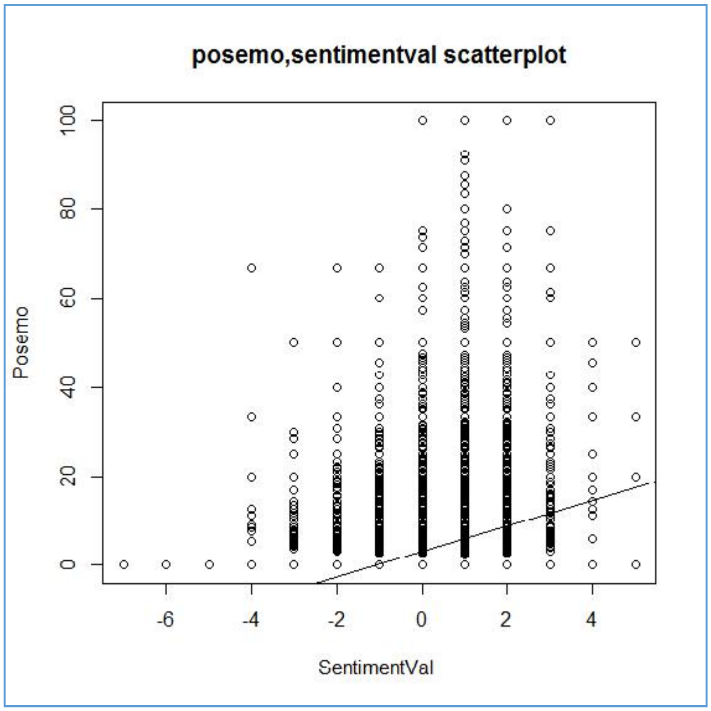


Figure 9 : Scatter plot of sentiment values with the posemo

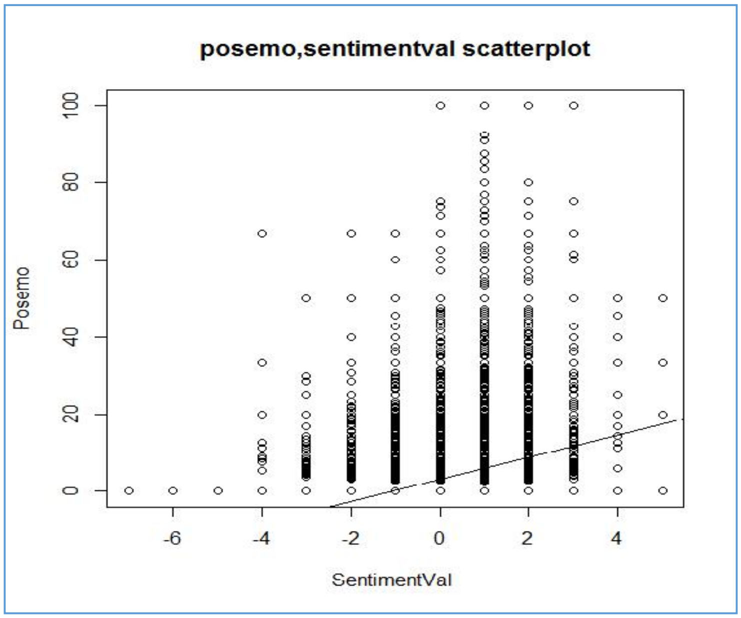


Figure 10 : Scatter plot of sentiment values with the negemo

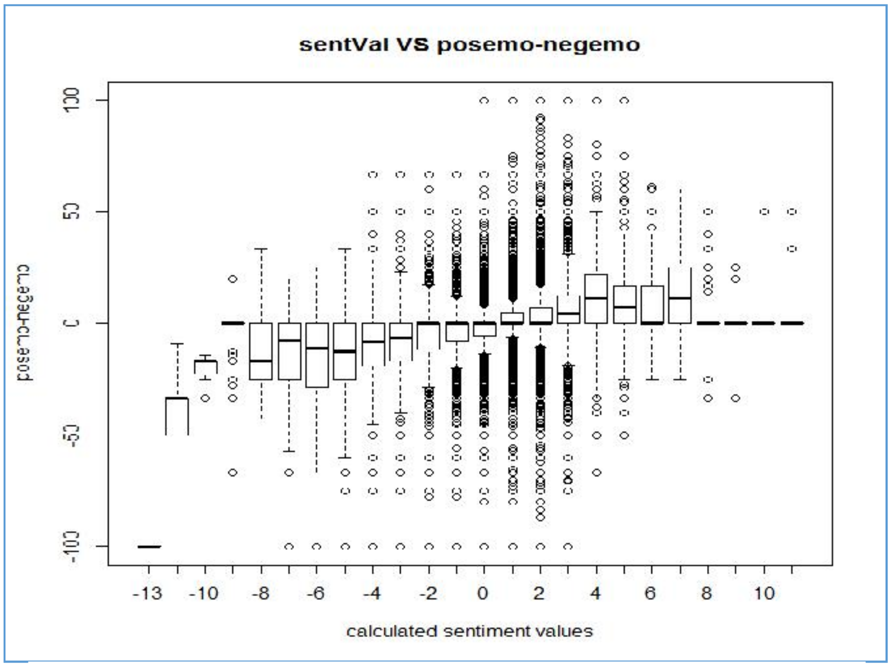
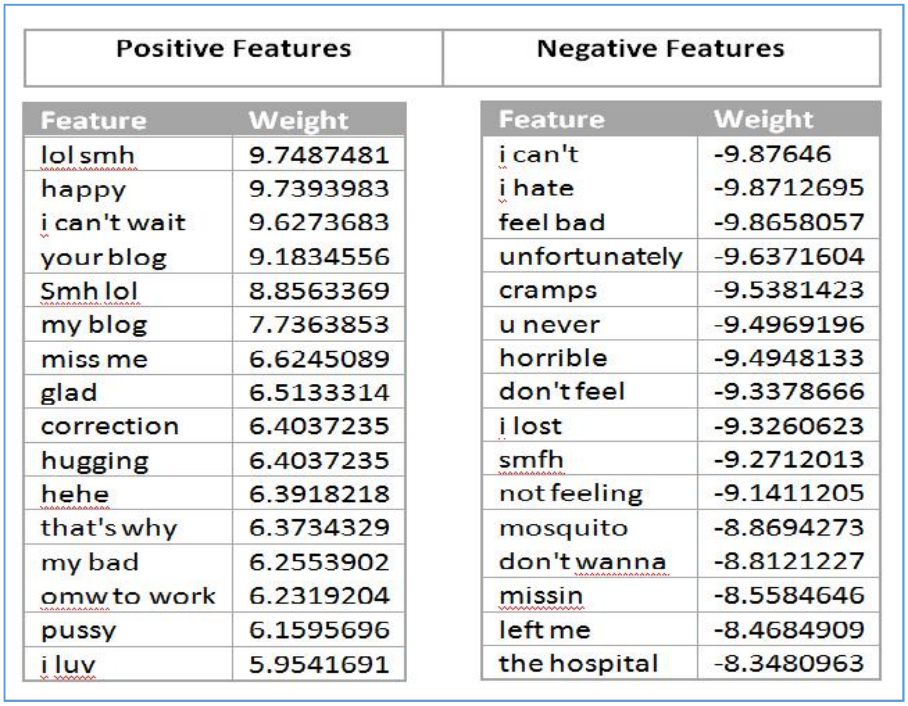


Figure 11 : Box plot of sentiment values and posemo,negemo

The data from sentiment dictionaries were processed to form single feature vectors and then they were classified by the classifier to check the intuitional precision of the classier. Below are the top 20 positive and negative feature identified.



Positive Features		Negative Features	
Feature	Weight	Feature	Weight
lol smh	9.7487481	i can't	-9.87646
happy	9.7393983	i hate	-9.8712695
i can't wait	9.6273683	feel bad	-9.8658057
your blog	9.1834556	unfortunately	-9.6371604
Smh lol	8.8563369	cramps	-9.5381423
my blog	7.7363853	u never	-9.4969196
miss me	6.6245089	horrible	-9.4948133
glad	6.5133314	don't feel	-9.3378666
correction	6.4037235	i lost	-9.3260623
hugging	6.4037235	smfh	-9.2712013
hehe	6.3918218	not feeling	-9.1411205
that's why	6.3734329	mosquito	-8.8694273
my bad	6.2553902	don't wanna	-8.8121227
omw to work	6.2319204	missin	-8.5584646
pussy	6.1595696	left me	-8.4684909
i luv	5.9541691	the hospital	-8.3480963

Figure 12 : Top 20 positive and negative single features

6 Conclusions

The experiments conclude that the model performed with decent precision levels, where the precision is found to be around 70.03% and recall of 68.04%. The calculated sentiment values seemed to a bit over fit as they had a large continuous range. An interesting observation is that the results have exhibited a fair amount of correlation value (0.39) with the posemo and negamo and almost negligent with the sad and anger values. The classifier performed extremely well with the single features as evident from the table show in the results.

7 References

1) Predicting Depression via Social Media. Munmun De Choudhury, Michael Gamon, Scott Counts, Eric Horvitz.

2) Modeling at Scale the Fine-Grained Dynamics of Mood. Chrisotpher Homan

3) Harnessing Twitter "Big Data" for automatic emotion identification. Wenbo Wang, Lu Chen, Krishnaprasad Thirunarayan, Amit P. Sheth

4) Sentiment in New York City: A High Resolution Spatial and Temporal View, Karla Z. Bertrand, Maya Bialik, Kawandeep Virdee, Andreas Gros and Yaneer Bar-Yam

5) Modeling Spread of Disease from Social Interactions. Adam Sadilek, Henry Kautz, Vincent Silenzio

6) The Development and Psychometric Properties of LIWC2007. James W. Pennebaker, Cindy K. Chung, Molly Ireland, Amy Gonzales, and Roger J. Booth

7) Support Vector Machines,

8) Amazon Mechanical Turks, http://aws.amazon.com/mturk/

9) SVM^{light} http://svmlight.joachims.org/