International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 07 | July 2020 www.irjet.net p-ISSN: 2395-0072

OPINION MINING USING TWITTER DATA SET

Gururaj S[1], Ayesha Sameen[2], Angana Prasad[3], Nida Tahreem[4]

[1]*Professor, Computer Science and Engineering Department, GNDEC Bidar, Karnataka, India*
[2,3,4]*Student, Computer Science and Engineering Department, GNDEC Bidar, Karnataka, India*

Abstract – *The project addresses the problem of sentimental analysis in twitter; that is classifying tweets according to the sentiments expressed in them: positive negative or neutral. Twitter is an online micro-blogging and social-networking platform which allows users to write short status updates of maximum length 280 characters. It is a rapidly expanding service with over 200 million registered users out of which 100 million are active users and half of them log on twitter on a daily basis- generating nearly 250 million tweets per day. Due to this large large amount of usage we hope to achieve a reflection of public sentiment is important by analyzing the sentiments expressed in the tweets. The main aim of this project is to develop a functional classifier for accurate and automatic sentiments classification of an unknown tweet stream.*

Key Words: **Sentiments, Tweets, Data Set, Twitter, Analysis, Emotion, Hashtag.**

1. INTRODUCTION

We have chosen to work with twitter since we feel it is a better approximation of public sentiment as opposed to conventional internet articles and web blogs. The reason is that the amount of relevant data is much larger for twitter, as compared to traditional blogging sites. Moreover the response on twitter is more prompt and also more general (since the number of users who tweet is substantially more than who write web blogs on a daily basis).

Sentiment analysis of the public is highly critical in macro-scale socioeconomic phenomena like predicting the stock market rate of a particular firm. This could be done by analyzing overall public sentiments towards that firm with respect to time and using economic tools for finding the correlation between public sentiment and the firm's stock market value. Firms can also estimate how well their product is responding in the market, which areas of the market is it having a favourable response and in which a negative response predicting the results of popular political election and polls is also an emerging application to sentimental analysis. One such study was conducted by tumasjan et al. in Germany for predicting the outcome of federal election which concluded that twitter is a good reflection of offline sentiment.

1.1 What is sentimental analysis?

Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral. Also referred to as opinion mining, it makes our goal to determine whether the data (tweet) is positive, negative or neutral. It is the automated process of identifying and extracting the subjective information that underlies a text. This can be either an opinion, a judgment, or a feeling about a particular topic or subject. The most common type of sentiment analysis is called 'polarity detection' and consists in classifying a statement as 'positive', 'negative' or 'neutral'.

1.2 Goal of the project:

The goal of the project was to predict the sentiments analysis can predict many different emotions attached to the texts, but in this project only 3 major was considered : positive, negative and neutral/ The training dataset was small (just over 5900 examples) and the data within was highly skewed, which greatly impacted on the difficulty of building good classifiers. After creating a lot of custom features, utilizing bag-of-words representation and applying the extreme gradient boosting algorithm, the classification accuracy at level of 58% was achieved.

2. PROBLEM STATEMENT

Be it any situation in today's world we can see there is a huge transformation going on to the digital world. In this transformation it is mandatory for us to upgrade ourselves as well. Gone are the days where a product will be launched and consumers /customers will come to shops or stores and tell you personally that what do they want or expect. There is a big transition to digital systems rather than physical presence. Especially we can see in the pandemic that almost everyone wants the things at their home sitting safe and secure.

In these kind of transformation we just can guess from those few who buy the product and bother to give you rating and manually product owner go to every statement and comment to check whether it's a sarcasm or a genuine rating out there. So in these situation we need automated systems to help us evaluate our own selves better.

Given a message, decide whether the message is of positive, negative, or neutral sentiment . For messages conveying both the positive and negative sentiment , whichever is the

strongest sentiment should be chosen and yeah, not to forget the sarcasm.

2.1 Project Scope

As we know the problem exist so the scope of solving the problem and overcoming the challenge remains vast and open ended until you don't find a perfect solution to fit in the place.

It can save a lot of man hours or we can say manual working hours which are just invested to check a thing again and again and then update the R&D team with the personal sentiments and bias. We can save a lot as well as rather then a personal bias the statements can be generalized based on figures and facts, rather that someone in control.

3. SYSTEM ANALYSIS

Feasibility Study:

The main objectives of the feasibility study are to determine whether the project would be feasible in terms of economic feasibility, technical feasibility and operational feasibility and schedule feasibility or not. It is to make sure that the input data which are required for the project are available. Thus we evaluated the feasibility of the system in terms of the following categories: Technical feasibility Operational feasibility Economic feasibility Schedule feasibility**.**

3.1 Technical Feasibility:

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at the point in time there is no detailed design of the system, making it difficult to access issues like performance, costs (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis; understand the different technologies involved in the proposed system. Before commencing the project, we have to be very clear about what are technologies that are to be required for the development of the new system. Is the required technology available? Our system is technically feasible since all the required tools are easily available. RStudio and Shiny makes the system more user and developer friendly and although all tools seem to be easily available there are challenges too.

3.2 Operation Feasibility:

Proposed project is beneficial only if it can be turned into information systems that will meet the operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? The proposed solution was to make a simplified web application. It is simpler to operate and can be used in any webpages. It is free and not costly to operate.

3.3 Economic Feasibility:

Economic feasibility attempts to weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system. A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case.

3.4 Schedule Feasibility :

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given period of time using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some projects are initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable. A minor deviation can be encountered in the original schedule decided at the beginning of the project. The application development is feasible in terms of schedule.

4.1 Study of Current System:

There are primarily two types of approaches for sentiment classification of opinionated text; Using a Machine learning based text classifier such as Naïve Bayes Using Natural Language Processing. We will be using those Machine learning and natural language processing for sentiment analysis of tweets.

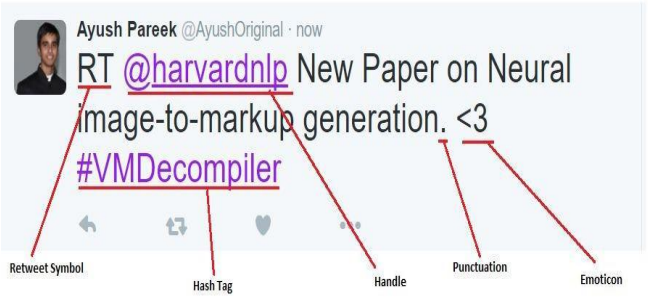
4.1.1 Machine Learning

The machine learning based text classifiers are a kind of supervised machine learning paradigm ,where the classifier need to be trained on some labelled training data before it can be applied to actual classification tasks. The training data is usually a extracted portion of the original data hand labelled manually. After suitable training they can be used on the actual test data. The Naive Bayes is a statistical classifier whereas Support Vector Machine is a kind of vector space classifier. The statistical text classifier scheme of Naive Bayes (NB) can be adapted to be used for sentiment classification problems as it can be visualized as a 2-class text classification problem: in positive and negative classes.

4.1.2 Pre-Processing

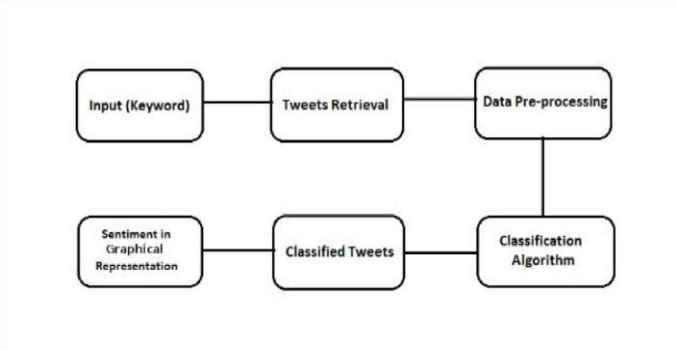
User-generated content on the web is seldom present in a form usable for learning. It becomes important to normalize the text by applying a series of pre-processing steps. We have applied an extensive set of pre-processing steps to decrease the size of the feature set to make it suitable for learning algorithms. Figure 2 illustrates various features seen in micro-blogging. Table 3 illustrates the frequency of

these features per tweet, cut by datasets. We also give a brief description of pre-processing steps taken.



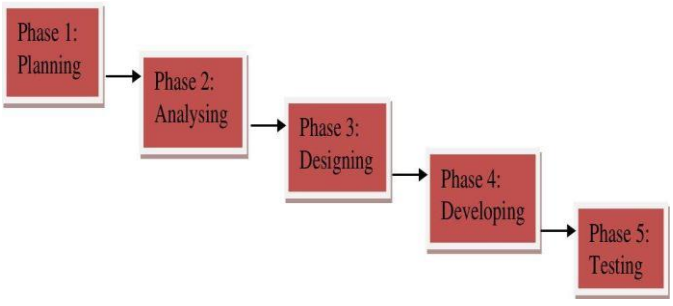
5. SYSTEM DESIGN

5.1 PROPOSED SYSTEM



The system was proposed that users enter the inputs i.e, Tweets/comments we retrieve those tweets on the basis of keywords. Further we do the pre processing steps to remove the unnecessary things like $, @ ETC. Based on the reliable algorithm we classify the inputs and once the data is classified we can have a proper sentiment representation in graphical form for even a layman to understand.

5.2 Development Life Cycle



There are five phases in this model and the first phase is the planning stage. It determine the objectives of the project whether the project should be given the green light to proceed. After approval, the next phase is analysis. Gathering and analyzing the system and user requirement is essential for entry to the design step. With the user requirements, the flow of the system is planned and the user interface is designed to suit there easy navigation needs.

After completing the design, actual coding begins databases are created and codes are written. With the development completed, testing will begin. The codes and databases are tested to ensure the result obtained. More time is spent on both development and testing stages because it is inevitable to have errors and issues and buffer time is allocated for troubleshooting.

6. SYSTEM DESIGN

The process of designing a functional classifier for sentiment analysis can be broken down into five categories. They are:

- Data Acquisition
- Human labelling
- Feature Extraction
- Classification
- Tweet/Mood Web Application

6.1 Data Acquisition

Data in the form of raw tweets is acquired by python library “tweetstream” which provides a package for simple twitter streaming API. This API allows two modes of accessing tweets: SampleStream and FilterStream.

FilterStream delivers tweets which match a certain criteria. It can filter the delivered tweets according to three criteria:

- Specific keyword(s) to track/search for in the tweets.
- Specific Twitter user(s) according to their user-id.
- Tweets originating from specific location(s) (only for geo-tagged tweets).

6.2 Human labelling

We gave the following guidelines to our labelers to help them in the labelling process:

- **Positive:** If the entire tweet has a positive/happy/excited/joyful attitude or if something is mentioned with positive connotations.
- **Negative:** If the entire tweet has a negative/sad/displeased attitude or if something is mentioned with negative connotations.

- **Neutral/Objective:** If the creator of tweet expresses no personal sentiment/opinion in the tweet and merely transmits information. Advertisements of different products would be labelled under this category.
- **Ambiguous:** If more than one sentiment is expressed in the tweet which is equally potent with no one particular sentiment standing out and becoming more obvious.
- **<Blank>:** Leave the tweet unlabeled if it belongs to some language other than English so that it is ignored in the training data.

6.3 Feature Extraction

There are some techniques which will aid us in Feature Extraction

Tokenization is a process of breaking a stream of texts into words into other meaningful words called "tokens".

Lowercase Conversion: Tweet may be normalized by converting it to lowercase which makes its comparison with an English dictionary easier.

Stemming: It is the text normalizing process of reducing a derived word to its root or stem.

Stop-words removal: Stop words are classes of some extremely common words which hold no additional information when used in a text and are thus claimed to be useless. Examples include "a", "an", "the", "he", "she", "by", "on", etc. It is sometimes convenient to remove these words because they hold no additional information.

Parts-of-Speech Tagging: POS-Tagging is the process of assigning a tag to each word in the sentence as to which grammatical part of speech that word belongs to, i.e. noun, verb, adjective, adverb, coordinating conjunction etc.

7. DATASET AND ALGORITHMS

Dataset:

The input data consisted of two CSV files: train.csv (5971 tweets) and test.csv (4000 tweets) - one for training and one for testing. Format of the data was the following (test data didn't contain Category column):

Id	Category	Tweet
635930169241374720	neutral	IOS 9 App Transport Security. Mm need to check if my 3rd party network pod supports it

7.1Bernoulli NB

Naïve Bayes Classifier for multivariate Bernoulli models like MultinomialNB works with occurrence counts , BernoulliNB is designed for boolean features.

7.2 Random Forest

Random forest is a supervised learning algorithm. It is used for classification problems. The algorithm creates decision trees on data samples and then get the prediction.

7.3 Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python is simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

8. IMPLEMENTATION

8.1 Processing Steps

I) Cleansing

● Remove URLs

● Remove usernames (mentions)

● Remove tweets with *Not Available* text

● Remove special characters

● Remove numbers

II) Text processing

● Tokenize

● Transform to lowercase

iii) Build word list for bag-of-words.

8.2 Sample Code

```
# plotly configuration plotly.offline.init_notebook_mode()

class TwitterData_Initialize():

data = []

processed_data = [] wordlist = []
```

data_model = None data_labels = None is_testing = False

def initialize(self, csv_file, is_testing_set=False, from_cached=None): if from_cached is not None:

self.data_model = pd.read_csv(from_cached)

return

self.is_testing = is_testing_set

if not is_testing_set:

self.data = pd.read_csv(csv_file, header=0, names=["id", "emotion", "text"]) self.data = self.data[self.data["emotion"].isin(["positive", "negative", "neutral"])]

else:

self.data = pd.read_csv(csv_file, header=0, names=["id",

"text"],dtype={"id":"int64","text":"str"},nrows=4000)

not_null_text = 1 ^ pd.isnull(self.data["text"]) not_null_id = 1 ^ pd.isnull(self.data["id"]) self.data = self.data.loc[not_null_id & not_null_text, :]

self.processed_data = self.data self.wordlist = []

self.data_model = None self.data_labels = None

data = TwitterData_Initialize() data.initialize("data\\train.csv") data.processed_data.head(5)

df = data.processed_data neg = len(df[df["emotion"] == "negative"]) pos = len(df[df["emotion"] == "positive"]) neu = len(df[df["emotion"] == "neutral"])

dist = [graph_objs.Bar(x=["negative","neutral","positive"], y=[neg, neu, pos],

)]

plotly.offline.iplot({"data":dist, "layout":graph_objs.Layout(title="Sentiment type distribution in training set")})

class TwitterCleanuper:

def iterate(self):

for cleanup_method in [self.remove_urls,

self.remove_usernames, self.remove_na, self.remove_special_chars, self.remove_numbers]:

yield cleanup_method

@staticmethod def remove_by_regex(tweets, regexp):

tweets.loc[:, "text"].replace(regexp, "", inplace=True) return tweets

def remove_urls(self, tweets):

return TwitterCleanuper.remove_by_regex(tweets, regex.compile(r"http.?://[^\s]+[\s]?"))

def remove_na(self, tweets):

return tweets[tweets["text"] != "Not Available"]

def remove_special_chars(self, tweets): # it unrolls the hashtags to normal words for remove in map(lambda r: regex.compile(regex.escape(r)), [",", ":", "\"", "=", "&", ";", "%", "$",

"@", "%", "^", "*", "(", ")", "{", "}",

"[", "]", "|", "/", "\\", ">", "<", "-",

"!", "?", ".", "'", "--", "---", "#"]):

tweets.loc[:, "text"].replace(remove, "", inplace=True) return tweets

def remove_usernames(self, tweets):

return TwitterCleanuper.remove_by_regex(tweets, regex.compile(r"@[^\s]+[\s]?"))

OUTPUT 1

Out[5]:

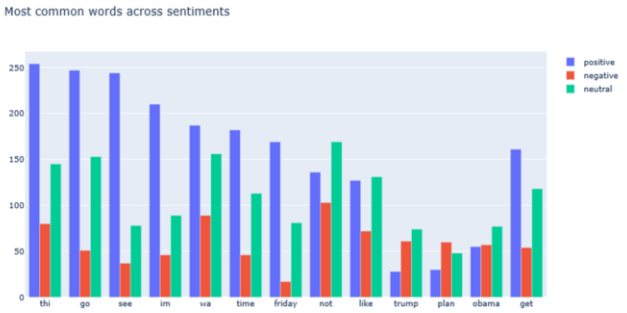
	id	emotion	text
0	635769805279248384	negative	Not Available
1	635930169241374720	neutral	IOS 9 App Transport Security. Mm need to check...
2	635950258682523648	neutral	Mar if you have an iOS device, you should down...
3	636030803433009153	negative	@jimmie_vanagon my phone does not run on lates...
4	636100906224848896	positive	Not sure how to start your publication on iOS?...

It's the first 5 elements of train data (used to train system) analysed by project.

OUTPUT 2

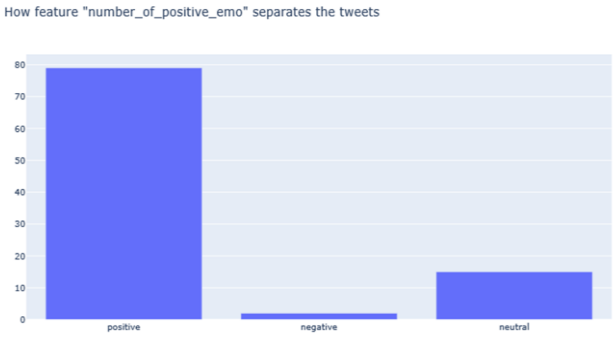
It tells the distribution that how much data in training were negative, positive or neutral.

OUTPUT 3



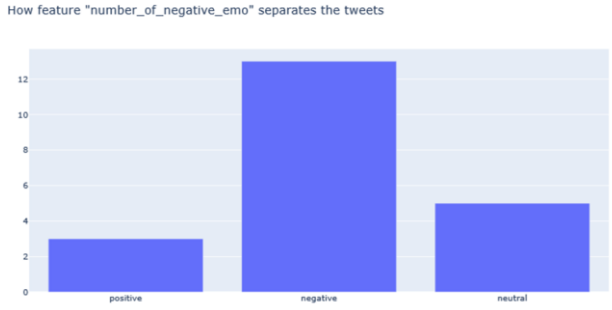
Graph to depict the distribution of most common words like the word thi was used in 254 positive tweets, 80 negative tweets and 145 neutral tweets

OUTPUT 4



Graph showing how the positive emoticons separate the tweets:- 79 were positive 2 were negative and 15 were neutral.

OUTPUT 5



Graph showing how negative emoticons separate/effect the tweet: - 3 positive 13 negative and 5 neutral.

OUTPUT 6

```
================================================
Testing RandomForestClassifier
Learing time 13.63689136505127s
Predicting time 0.4843587875366211s
=================== Results ====================
            Negative     Neutral     Positive
F1       [0.24362606 0.47313692 0.69605037]
Precision[0.4673913  0.4806338  0.62874871]
Recall   [0.16475096 0.46587031 0.77948718]
Accuracy 0.5679164105716041
================================================
```

The output of theorem Random Forecast Classifier.

TESTING

Unit Testing

Unit testing focuses verification efforts on the smallest unit of the software design, the module. This is also known as "Module Testing". The modules are tested separately. This testing carried out during programming stage itself. In this testing each module is found to be working satisfactorily with regards to the expected output from the module.

Integration Testing

Data can be shared across the modules; one module can have adverse efforts on another. Integration testing is a systematic testing for constructing the program structure while at the same time conducting tests to uncover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here, correction is difficult because the isolation of cause is complicated by the vast expense of the entire program. Thus, in the integration testing phase, all the errors uncovered are corrected for the next testing steps.

System Testing

System testing is the stage of implementation that is aimed at ensuring that the system works accurately and efficiently for live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, then goal will be successfully achieved.

Output testing

The next step is output testing of the proposed system since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated by the system under consideration. Here, the output format on the screen is found to be correct as per the format which was designed in the system designing phase according to the user needs.

ADVANTAGES

- No human interaction to guess and judge the comments/feedbacks.
- No personal bias affecting the final outcome.
- Ease to analyze thousands of data set just with a click. Little to no maintenance.
- As the dataset increase so does the accuracy to correct efficiently.
- No expenditure of capital on the required task as compared to manual job.

CONCLUSION

We conclude that using different NLTK classifier it is easier to classify the tweets and moreover we improve the training data set to give accurate results.

The task of sentiment analysis, especially in the domain of micro-blogging, is still in the developing stage and far from complete. So we propose a couple of ideas which we feel are worth exploring in the future and may result in further improved Performance. Right now we have worked with only the very simplest unigram models; we can improve those models by adding extra information like closeness of the word with a negation word. We could specify a window prior to the word (a window could for example be of 2 or 3 words) under consideration and the effect of negation may be incorporated into the model if it lies within that window. The closer the negation word is to the unigram word whose prior polarity is to be calculated, the more it should affect the polarity. For example if the negation is right next to the word, it may simply reverse the polarity of that word and farther the negation is from the word the more minimized ifs effect should be. Apart from this, we are currently only focusing on unigrams and the effect of bigrams and trigrams may be explored. As reported in the literature review section when bigrams are used along with unigrams this usually enhances performance.

BIBLOGRAPHY

[1] Albert Buffet and Eibe Frank. Sentiment Knowledge Discovery in Twitter Streaming Data. Discovery Science, Lecture Notes in Computer Science, 2010, Volume 6332/2010, 1-15, DOI: 10.1007/978-3-642-16184-1_1

[2] Alec Go, Richa Bhayani and Lei Huang. Twitter Sentiment Classification using Distant Supervision. Project Technical Report, Stanford University, 2009.

[3] Alexander Pak and Patrick Paroubek. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In Proceedings of international conference on Language Resources and Evaluation (LREC), 2010.

[4] Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner and Isabell M. Welpe. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM), 2010. Project Thesis Report 51

[5] Bo Pang, Lillian Lee and Shivakumar Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2002.

[6] Chenhao Tan, Lilian Lee, Jie Tang, Long Jiang, Ming Zhou and Ping Li

[7] Efthymios Kouloumpis, Theresa Wilson and Johanna Moore. Twitter Sentiment Analysis: The Good the Bad and the OMG! In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM), 2011.

[8] Hatzivassiloglou, V., & McKeown, K.R.. Predicting the semantic orientation of adjectives. In Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL, 2009.

[9] Johan Bollen, Alberto Pepe and Huina Mao. Modelling Public Mood and Emotion: Twitter Sentiment and socio-economic phenomena. In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM), 2011.