‘Meme’tic Engineering to Classify Twitter Lingo

Priyashree S.[1], Shivani N.[2], Vigneshwar D.K.[3], Karthika S.[4]
Department of Information Technology
SSN College of Engineering
Chennai, India
skarthika@ssn.edu.in

Abstract—**Text memes have become a popular form of communicating ideologies on social networking websites. Some of these have led to drastic consequences in the society and has also influenced lives of several people. Text memes are pieces of text that are copied and spread rapidly by internet users. The sentiment can refer to two different types: emotions and opinions. These opinions can be divided into three classes: positive, negative and neutral. The inference obtained from this paper are the various categories of text memes circulated amongst users and an analysis of the over-all reaction or sentiment of the public.**

Keywords— emotions; opinions; sentiment;

I. Introduction

In the last couple of years, the social medium Twitter has become more and more popular. Since Twitter is the most used microblogging website with about 500 million users and 340 million tweets a day, it is an interesting source of information. The messages, or in Twitter terms the tweets, are a way to share interests publicly or among a defined group. Twitter is widely adopted through all strata, it can be seen as a good reaction of what is happening around the world.

In this research work, machine learning approach and natural language processing techniques have been used to understand the patterns and characteristics of tweets and predict the sentiment (if any) they carry. Specifically, we build a computational model that can classify a given tweet as either positive, negative or neutral based on the sentiments it reflects. A positive and negative class would contain polar tweets expressing a sentiment. However, a neutral class may contain an objective or subjective tweet either a user reflect neutrality in an opinion or contain no opinion at all.

A. ‘Meme’tic Engineering:

A MEME is an idea that spreads like a virus by word of mouth, email, blogs, etc. This paper focuses on text memes which are pieces of text, copied and spread rapidly by internet users. ‘Meme’tic engineering refers to the process of handling memes using various methods such as searching, classifying, processing, etc[8].

B. Sentiment Analysis

Sentiment analysis involves in several research fields like, natural language processing, computational linguistics and text analysis. It refers to the extraction of subjective information from raw data, often in text form which consist of different kinds of sentiments. The sentiment can refer to opinions or emotions. These two types are slightly related but there is an evident difference. In sentiment analysis based on opinions, a distinction is made between positive and negative opinions. On the other hand, sentiment analysis based on emotions, is about the distinction between different kinds of emotions. The sentiment analysis that is considered in this study is based on opinions and is often referred to as opinion mining. Sentiment analysis aims to determine the attitude of the opinion holder with respect to a subject. This application also determines the overall sentiment of a topic.

II. Literature Review

A. Twitter Sentiment Analysis using Machine Learning

In the study ‘Thumbs up? Sentiment Classification using Machine Learning Techniques’, the documents are classified by overall sentiment rather than by topic. The Machine Learning techniques employed by them include I Bayes, maximum entropy classification, and support vector machines. Movie reviews have been grouped into positive and negative based on individual connotations. This gives a clear understanding of making use of appropriate algorithms for the required application [4]. The study handles only positive and negative sentiments and does not explore those statements that fall under the neutral category.

B. Twitter Sentiment Analysis:

In the study, “Twitter sentiment analysis: The good the bad and the omg!” investigation of the linguistic features has been carried out for detecting the sentiment of twitter messages. Research has been performed to evaluate the usefulness of existing lexical resources as well as features that capture information about the informal and creative language used in microblogging. The conclusion derived from the study shows that the parts-of-speech method may not be useful for sentiment analysis in the microblogging domain[3].

This particular study has only dealt with Twitter hashtags and emoticons. The datasets used, have not considered the complete tweet. This fails to process the overall sentiment of the tweet.

C. Handling Datasets

Eight publicly available and manually annotated datasets have been evaluated in the report, “Evaluation datasets for Twitter sentiment analysis: A survey and a new dataset, the STS-Gold” to assess the performance of sentiment analysis on Twitter. STS-Gold is a new evaluation dataset that has been created where tweets and targets (entities) are annotated individually and therefore may present different sentiment

978-1-5090-5595-1/17/$31.00 ©2017 IEEE

labels. This paper also provides a comparative study of the various datasets along several dimensions including: total number of tweets, vocabulary size and sparsity which enables efficient procedures for sentiment analysis[5].

III. Proposed System

A. Twitter Lingo Classifier

The Twitter Lingo analysis starts with data collection using Twitter API, tweepy which acts as an interface between Python and Twitter. The data is searched using a keyword entered by the user. The gathered data is then sent to the pre-processing unit where the tweets are cleaned by eliminating stop words apart from converting to lowercase, tokenizing and limiting the language to English only. The classification of the pre-processed data is carried out using different algorithms in Python.

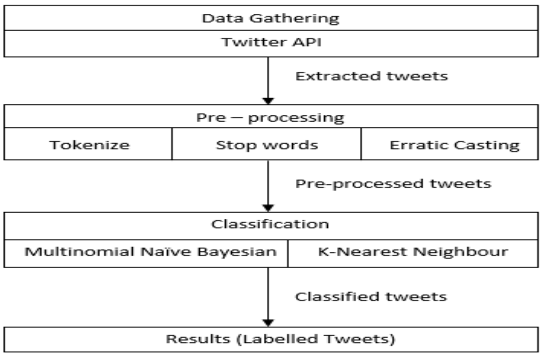

Figure 1: Flow of proposed system

B. Algorithms:

This part deals with the different methodologies employed in order to classify the tweets in the Twitter Lingo Classifier.

1. Naïve Bayesian Classifier

The I Bayesian classifier is based on Bayes' theorem with independence assumptions between predictors. A I Bayesian model is easy to build, with no complicated iterative parameter estimation which makes it particularly useful for very large datasets. Despite its simplicity, the I Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods[6].

Bayes theorem provides a way of calculating the posterior probability, P(c|x), from PI, P(x), and P(x|c). I Bayes classifier assume that the effect of the value of a predictor (x) on a given class I is independent of the values of other predictors. This assumption is called class conditional independence.

$$P(c|x)= (P(x|c)\times PI)\div P(x) \qquad (1)$$
$$P(c|X)= P(x_1 |c) \times P(x_2 |c) \times \ldots. P(x_n |c) \times PI \qquad (2)$$

where,
P(c|x) is the posterior probability of class (target) given predictor (attribute).
PI is the prior probability of class.
P(x|c) is the likelihood which is the probability of predictor given class.
P(x) is the prior probability of predictor.

There are two I Bayes' models: Multinomial and Bernoulli. The multinomial I Bayes model generates one term from the vocabulary in each position of the document. The Bernoulli model generates an indicator for each term of the dataset. The value 1 indicates that a term is present and a 0 indicates absence. The Bernoulli model does not count the multiple occurrence of terms whereas multinomial I Bayes does. This work uses Multinomial naïve Bayes classifier since it performs faster than other classifiers apart from weighing features independently.

2. K-Nearest Neighbours Classifier

In pattern recognition, the k-nearest neighbours algorithm (k-NN) is a non-parametric method used for classification. The input consists of the k closest training examples in the feature space.

In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor[7].

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. In the classification phase, k is a user-defined constant, and an unlabelled vector (a query or test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point. A commonly used distance metric for continuous variables is Euclidean distance. For discrete variables, such as for text classification, another metric can be used, such as the overlap metric (Hamming distance). In text classification nearest neighbour achieves reasonable results in combination with the tf-idf measure.

The application utilises 'KneighboursClassifier' from sklearn package available on Python. An observation that the maximum accuracy was obtained using this classifier because:

- Finding the nearest neighbours in a smaller dataset is much easier
- It uses a similarity measure to classify new cases
- The similarity between the training and test dataset will be effective in predicting the class of the data

C. Datasets

The datasets used are a collection of tweets and are elaborately discussed in the following section.

1. Benchmark Dataset

Table 1 shows a sample of the benchmark dataset which consists of 1524 tweets comprising of a combination of positive, negative and neutral labels. This dataset is a Comma Separated (CSV) file and was extracted from generic benchmark datasets [5].
Dataset: twitter-sanders.csv

TABLE 1: BENCHMARK DATASET

Label	Tweet
1	@londonbridget13 ill always have your back!!
1	ha im soo obsessed with taylor swift's album she just soo talented
1	Whirlpool Galaxy Deep Field. what an amazing universe
-1	Feels sad my best friend Skippy is not going to be in London Thanks for ruining that T! ;)
-1	@Shari58 Still got the fever
-1	I hate having my eyes dialated....I have a headache
0	I help it & explored the #iphone4s @apple store & talked to #Siri... Should've tried to speak to her in French?! #nexttime
0	@Safari I should probably try it out
0	This is a speech by my favourite President. #checkitoutsoon

2. Dynamic dataset using Twitter API

The tweets are gathered at real time using the search function available in tweepy[12]. The access tokens and consumer keys have been used to connect the Twitter handle. This helps in retrieving various tweets based on a particular search keyword entered by the user. The collected tweets are stored in a csv file, then pre-processed and classified. Table 2 shows a sample set of tweets retrieved dynamically on entering the keyword 'demonetization'.

TABLE 2: DYNAMIC DATASET

Tweet
b"RT @GauravPandhi: It's time for another demonetization drive?"
b"RT @shabanais: This is the situation in all ATM's in my area three months after demonetization. This is South Delhi. Imaginesituation in\xe2\x80\xa6"
b'RT @Vishj05: Lets have the uh so successful demonetization drill again then and fight Black money.'

D. Representation of system

The functionality of the application is represented using a sequence of steps with a basic pseudocode.

```
Basic Pseudocode:
DO
INSTALL the required packages
INPUT the DATASET, d
INITIALIZE data onto CSV file
END

DO
IMPORT required packages
For each tweet in d
PERFORM classification using 'MultinomialNB', 'KNeighbours', 'LogisticRegression' and 'SVC'
CLASSIFY as
        +1 for Positive
        -1 for Negative
        0 for Neutral
DISPLAY classified tweet
VISUALIZE using pie-chart showing results
CALCULATE accuracy, A
$A= (TP+TN)/(TP+TN+FP+FN)$
where
        TP is True Positive,
        TN is True Negative,
        FP is False Positive and
        FN is False Negative respectively
CALCULATE Precision, P
$P=TP/(TP+FP)$
CALCULATE Recall, TPR
$TPR=TP/(FN+TP)$
CALCULATE f-score,f
$f=2(P*TPR/(P+TPR))$
END
END
```

E. Estimation Measures

The different performance evaluation metrics are:

- Accuracy - It is a measure of statistical bias and a description of systematic errors. It is given as:
  $$\text{Accuracy} = (TP+TN)/\ (TP+FP+TN+FN) \quad (3)$$
- Precision - Proportion of instances that are truly of a class divided by the total instances classified as that class also called Positive Predictive Value(PPV)
  $$PPV = TP/\ (TP+FP) \quad (4)$$
- Recall - proportion of instances classified as a given class divided by the actual total in that class (equivalent to TP rate)
  $$\text{Recall} = TP/\ (TP+FN) \quad (5)$$
- F-Score - A combined measure for precision and recall calculated as
  $$F = (2*PPV*\text{Recall})/\ (PPV+\text{Recall}) \quad (6)$$

The different estimation measures have been analysed and tabulated for the various algorithms used. From Table 3, using Eq.3, it is observed that the accuracy for K-Nearest Neighbours (KNN) classifier is the highest followed by Multinomial naïve Bayes (MNB) classifier. MNB works well when compared to SVM or Logistic Regression because it is not just a two-class classifier like SVM. And, each feature is weighed independently unlike LR and does not over fit data. MNB is a generative model since it selects the most likely features from the given data.

TABLE 3: ACCURACY

Algo/ Label	Support Vector Machine	Logistic Regression	Multinomial Algorithm	K-Nearest Neighbours Algorithm
Benchmark set	31.80%	72.45%	73.77%	62.29%
Dynamic set	38.77%	54.08%	65.30%	66.32%

Table 4 shows that the precision is observed to be high for the positive label when the algorithms LR and MNB are applied to the dynamic dataset. But it is seen that SVM shows 0.0 for positive and neutral classes since the data is over fitted. Such an overfitting of data happens when a function is closely fit to a limited set of data points.

TABLE 4: PRECISION

Algorithm / Label	Support Vector Machine	Logistic Regression	Multinomial Algorithm	K-Nearest Neighbours Algorithm
Positive	0.0	78.57%	83.67%	70.47%
Negative	31.80%	63.71%	61.90%	61.11%
Neutral	0.0	76.59%	80.24%	56.25%

Table 5 shows that the recall is observed to be high for the negative label when the algorithms LR and MNB are applied to the dynamic dataset. This is because the more relevant queries are fitted under the negative label and their sensitivity is higher.

TABLE 5: RECALL

Algo/ Label	Support Vector Machine	Logistic Regression	Multinomial Algorithm	K-Nearest Neighbours Algorithm
Positive	0.0	72.64%	77.35%	69.81%
Negative	1.0	74.22%	80.41%	45.36%
Neutral	0.0	70.58%	63.72%	70.58%

Tables 6 shows the f-score percentages respectively for the three class labels along with the classification methodologies used for the dynamic dataset. F-score gives the combined result obtained from precision and recall.

TABLE 6: F-SCORE

Algo / Label	Support Vector Machine	Logistic Regression	Multinomial Algorithm	K-Nearest Neighbours Algorithm
Positive	0.0	75.49%	80.39%	70.14%
Negative	48.25%	68.57%	69.95%	52.07%
Neutral	0.0	73.46%	71.03%	62.60%

F. Platforms and packages used:

- Python 2.7
- Numpy
- Flask, NLTK, Tweepy, Pandas (Python Library)

IV. RESULTS

The various processing techniques and their experimental results that were recorded during the development of this application are presented in this section.

Pre-processing

Original Tweet:
@myindmakers: Modi has reenergized the party cadres with a resounding rebuttal in the Parliament. https://t.co/81chOtjdtT'
Pre-processed Tweet:
modi has reenergized the party cadres with a resounding rebuttal in the parliament

Tokenization: It is the process of breaking up a stream of text into words, symbols and other meaningful elements called “tokens”. Tokens can be separated by whitespace characters and/or punctuation characters. It is done so that we can look at tokens as individual components that make up a tweet.

URL’s and user references (identified by tokens “http” and “@”) are removed since we are interested in only analysing the text of the tweet.

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

Stop-words removal: Stop words are class 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 since they are used almost equally in all classes of text.

Classification of tweets

Three different output cases have been discussed in order to distinguish between the positive, negative and neutral results[9,10].

Search Case 1:

The user enters the required search keyword, based on which the Twitter API collects the corresponding set of Tweets. The keyword ‘rahul dravid’ has been entered in order to retrieve the relevant data. Figure 2, shows the pie-chart which provides the visualization of the overall sentiment on the topic ‘rahul dravid’. The general feeling on this topic is observed to be positive [10]. The classified set of tweets have also been incorporated in the output with 1 denoting a positive tweet, -1 for negative and 0 for neutral [1,2]

TABLE 7: DYNAMIC TWEETS FOR KEYWORD ‘RAHUL DRAVID’

Label	Tweet
1	rahul dravid amp zaheer khan two of the most loved cricketers ipl ipl10 rahuldravid dd
1	at 38 rahul dravid is called old and see what wonders he performs at 40 rahul gandhi is called young and see what bxe2x80xa6
1	b u are the 2nd wall of cricket after respected 1st wall of cricket rahul dravid

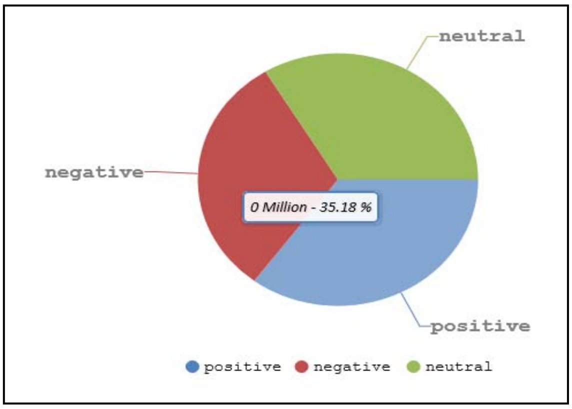

Figure 2: Result Positive

Search Case 2:
The keyword ‘osama’ was entered and the result obtained is denoted in Figure 3. The general feeling on this topic is observed to be negative.

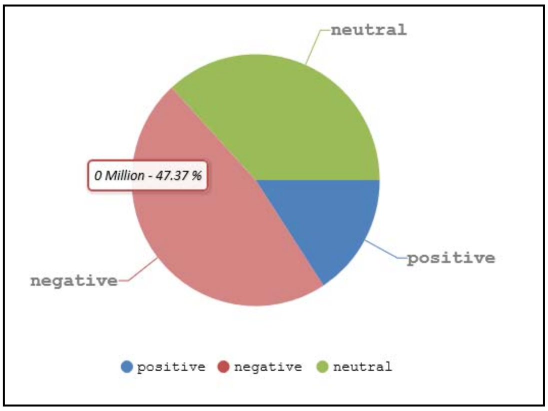

Figure 3: Result Negative

TABLE 8: DYNAMIC TWEETS FOR KEYWORD ‘OSAMA’

Label	Tweet
-1	israeli occupation forces storm cartoonist osama nazzals home wreak havoc and damage his art
-1	guy222 ahmadkhan world knows u r the exporter of terrorismosama was in abottabad and dawood is in karachi
-1	i refuse to believe one of these killed osama bin laden

Search Case 3:
The keyword ‘demonetization’ obtained an overall neutral reaction as denoted in Figure 4. The general feeling on this topic is observed to be neutral.

TABLE 9: DYNAMIC TWEETS FOR KEYWORD ‘DEMONETISATION’

Label	Tweet
0	i liked a video from adsensegate the truth about youtube demonetization
0	disruptive policy innovation against corruption demonetization in india by compliancexe2x80xa6
0	how does that explain the screen shots wsj took of a coke ad months after the demonetisationxe2x80xa6

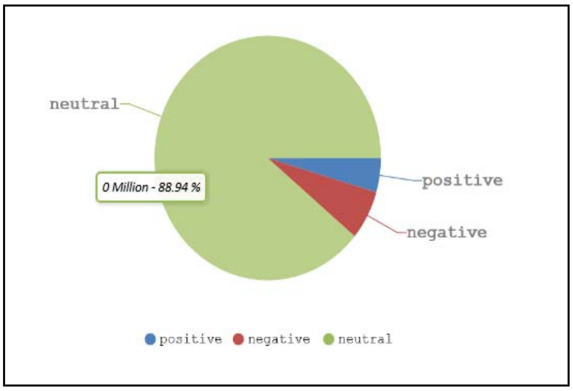

Figure 4: Result Neutral

V. CONCLUSION

Twitter is a platform where highly unstructured, short messages are available. This paper successfully classifies tweets based on a search topic and analyses the overall impact that an issue has on the society. Most companies and businesses these days focus on evaluating the sentiments of people on various issues. This study presents a small-scale working application of such an analyser and studies its functionalities. Classification of text memes into positive, negative and neutral gives us a clearer perspective of reactions of the general public on different areas. The KNN and MNB classifier algorithms prove to be accurate in making the necessary classifications in this application. Thus, the application has devised an efficient sentiment evaluator which is in demand today in all enterprises.

REFERENCES

[1]. Cervesato, Iliano, et al. "2009 Senior Thesis Project Reports.", 2010.
[2]. Groot, R. de. Data mining for tweet sentiment classification. MS thesis, 2012.
[3]. Kouloumpis, Efthymios, Theresa Wilson, and Johanna D. Moore. "Twitter sentiment analysis: The good the bad and the omg!." ICWSM 11.538-541, 2011.
[4]. Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.
[5]. Saif, Hassan, et al. "Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the sts-gold.", 2013.
[6]. Thakkar, Mr Harsh Vrajesh. "Twitter Sentiment Analysis using Hybrid Naïve Bayes."
[7] Scikit-learn.org, ‘K-Nearest Neighbours’, 2016, [Online] Available:http://scikit-learn.org/stable/modules/neighbors.html [Accessed:November - 2016]
[8] Memeburn.com, ‘Words used in Social Media’, 2016, [Online] Available: https://memeburn.com/2010/09/text-mining-reveals-best-and-worst-words-used-in-social-media/ [Accessed:December - 2016]
[9] Python.org, 'Pandas’, 2017, [Online]. Available: pypi.python.org/pypi/pandas/ [Accessed: December - 2016]
[10] Jetbrain.com, ‘PyCharm’, 2016, [Online] Available: www.jetbrains.com/pycharm/ [Accessed: December – 2016]
[11] Twitter.com, 'Twitter Application’, 2017, [Online]. Available: dev.twitter.com/ [Accessed: November - 2016]
[12] Twitter.com, 'Twitter Application 2’, 2017, [Online]. Available:apps.twitter.com/ [Accessed: November - 2016]