NOVEL APPROACH FOR EMOTION DETECTION USING TEXT

Tannishk Sharma[1] , Mohammad Zeb Alam[2] , Harjot Singh Bhatia[3]
[1] Department of Information Technology, BVCOE Delhi
[2] Department of Information Technology, BVCOE Delhi
Department of Information Technology, BVCOE Delhi

Abstract -- Emotions are described as a person's feeling and his mood. It has been studied deeply in psychology and related sciences. Emotions can be expressed in terms of body language, facial expressions and written medium. Emotion detection is considered part of analysis of human Psyche . The analysis finds the nature of the text to assign it to a category. Emotions can be classified into sadness, love, fear, anger, joy and surprise. Textual data can reflect the emotional state of the person. There can be various applications of textual analysis for better understanding of the consumer. Using textual analysis can be done by keyword technique, linguistic rule based technique or by machine learning technique.

This paper focuses on implement and study of different classification algorithms for detection and analysis of Human Emotions using Text . Later in the paper we would calculate different set of emotions for live data using our consolidated classification approach . Our first analysis of accuracy of different algorithms is carried on UCI dataset . The opinion shared by users also have their emotional state within the text, using classification we can use it for data mining. A major difficulty faced is the interpretation of slangs as Twitter uses only 140 character limit for sharing opinion. The project also includes creating platform to check live tweets and show accuracy of each algorithm with the result of each.

Keywords : Emotion , SVM , Naïve Bayes , Consolidated Classifier , Logistic Regression

I. INTRODUCTION

Emotions are described as a person's feeling and sways with mood. It has been studied deeply in psychology and related sciences. Emotions can be expressed in terms of body language, facial expressions and written medium. Emotion detection is considered part analysis of the human Psyche . The analysis finds the nature of the text to assign it to a category. Emotions can be classified into ***sadness, love, fear, anger, joy*** and ***surprise***. Textual data can reflect the emotional state of the person. There can be various applications of textual analysis for better understanding of the consumer. Textual analysis can be done by a keyword technique-a linguistic rule based technique or by machine learning.

This report focuses on implementing different classification algorithms for detection and classification of accuracy of emotions using naive bayes, Logistic regression, SVM (Support vector Machine) and consolidated classifying approach and and check accuracy of different algorithms. It would calculate different set of emotions on live data. The opinion shared by users also have their emotional state within the text, using classification we can use it for data mining. A major difficulty faced is the interpretation of slangs as Twitter uses only 140 character limit for sharing opinion. The project also includes creating platform to check live tweets using consolidated approach .

Recent development in sentimental analysis has increased interest in computational linguistics. Emotion involved in computer interaction with humans has been stated by Since Picard in The concept of affective computing.

Keyword based technique uses algorithm to spot and match different keywords that indicate to a certain emotion class. However this simple method cannot be used for complex and large textual data. Lexical rule based techniques use probabilities that are assigned to various words.

This technique fails to analyze emotional content that might not be on keywords. Thus, raising a need for a methodology of certainty-Supervised and unsupervised learning techniques have been used in machine learning to classify textual data for a predictive emotional analysis to entrench disjoint data over a class of processes developed for intended research.

II.RELATED WORK

Sentiment Analysis is in itself becoming a major area of study un- der Machine learning. The ideology used in this project is based on the underlying principles developed in [5] where the tweets were classified using unigram vectors and training was performed by distant supervision. The research in [9] elucidates that the use of emoticons as labels is effective in reducing dependencies in ma- chine learning. The analysis in [9] is also on the basis of a query term and feature reduction using algorithms like Naive Bayes, Maximum Entropy and Support Vector Machines. The research and analysis conducted by Pang and Lee [9] was used to analyze the performance of different machine learning techniques in the movie review domain. It has also found implementations [10] as a sub component technology in augmentation with other systems like emails and online advertisements. With the help of improved Natural Language Processing capabilities and tools, this domain is gaining widespread importance and improved application in various other fields.

III.TECHNIQUES

3.1 Naive Bayes

In machine learning, **naive Bayes classifiers** are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

Naive Bayes has been studied extensively since the 1950s. It remains a popular (baseline) method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the features. With appropriate pre-processing, it is competitive in this domain with more advanced methods including support vector machines.It also finds application in automatic medical diagnosis.

Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

In the statistics and computer science literature, Naive Bayes models are known under a variety of names, including **simple Bayes** and **independence Bayes**.All these names reference the use of Bayes' theorem in the classifier's decision rule, but naive Bayes is not (necessarily) a Bayesian method.

A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

For some types of probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods.

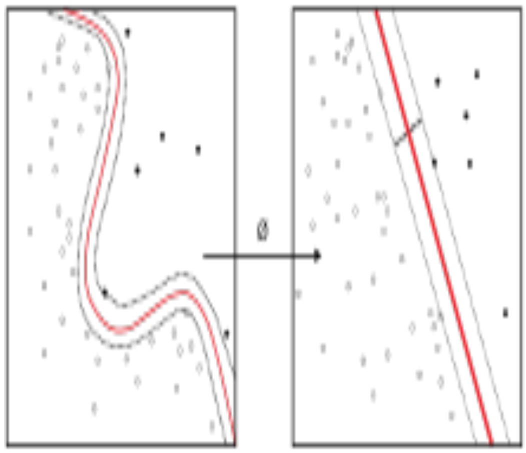
Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations. In 2004, an analysis of the Bayesian classification problem showed that there are sound theoretical reasons for the apparently implausible efficacy of naive Bayes classifiers.

3.2 SVM

Support vector machine

SVM is a supervised learning technique used in machine learning, this is done to analyze data. With a set of training examples, belonging to either category. SVM training algorithm to build a technique to place new examples to either category. The examples are mapped in two categories with a boundary as wide as it may. New data is mapped and placed in the specified category.

For not labeled data unsupervised learning is used, which attempts to find a natural clustering of the data to groups. Thus, new data is mapped to these formed groups. The clustering algorithm which provides an improvement to the support vector machines is called **support vector clustering [2].**

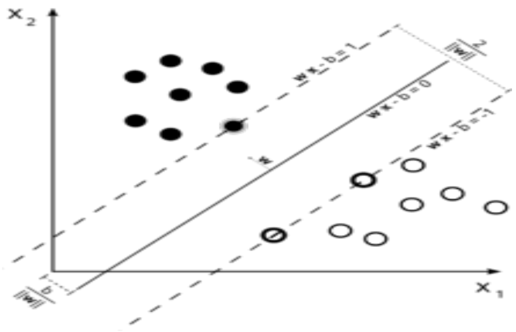


Training dataset of n points of the form

(vec(x,1),(y,1)),.......,(vec(x ,n) , (y,n))

where the y(i) are either 1 or −1, each indicating the class to which the point vec(x(i)) belongs. Each vec(x(i)) is a p-dimensional real vector. We want to find the "maximum-margin hyperplane" that divides the group of points vec(x(i)) for which y(i)=1 from the group of points for which y(i) = -1, which is defined so that the distance between the hyperplane and the nearest point vec(x(i)) from either group is maximized.[1]

A Hyper plane written as the set of points vec(x(i))[1]



vec(w).vec(x)+b=0,

where,

vec(w) is the normal vector to the plane.

b/||vec(w)|| is offset of the hyper plane.

Broadly two Types:

-Hard Margin.

-Soft Margin

3.3 Logistic Regression
Logistic regression, despite its name, is a linear model for classification rather than regression. Logistic regression is also known in the literature as logit regression, maximum-entropy classification (MaxEnt) or the log-linear classifier. In this model, the probabilities describing the possible outcomes of a single trial are modeled using a logistic function.
The implementation of logistic regression in scikit-learn can be accessed from class **LogisticRegression**. This implementation can fit binary, One-vs- Rest, or multinomial logistic regression with optional L2 or L1 regularization.
As an optimization problem, binary class L2 penalized logistic regression minimizes the following cost function:

$$\min_{w,c} \frac{1}{2}w^T w + C \sum_{i=1}^{n} \log(\exp(-y_i(X_i^T w + c)) + 1).$$

Similarly, L1 regularized logistic regression solves the following optimization problem

$$\min_{w,c} \|w\|_1 + C \sum_{i=1}^{n} \log(\exp(-y_i(X_i^T w + c)) + 1).$$

The solvers implemented in the class **LogisticRegression** are “liblinear”, “newton-cg”, “lbfgs” and “sag”:

3.4 Consolidated Classifier Approach
This is our novel classifier approach in which we use linear combination of the above specified classifiers. It has 2 variants :

- Weighted : Where all classifiers are given weight according to their accuracy . Category with max score is selected
- Non Weighted : Where all classifiers are given equal weights and category with max weight is selected .

If weights of 2 categories are equal then one to which classifier with maximum efficiency belong is selected .

Weights can also be according to F Score , Precision , Recall etc

IV.DATASETS

The whole process of data sets is divided into two steps in which one where UCI machine learning repository is used to collect a finite set of data of texts and emotions based on them , indeed which is used for preliminary training of the classifier. The trained classifier is put forth for further testing, and used to estimate the accuracy for the classifier based on the same dataset .

In the second step , we built an the platform or application works over the data of Twitter, which is extracted using a twitter search API wrapper in python. The following extracted data is corroborated over the search keywords.

TRUMP
DONALD TRUMP

This array of search keywords are tested over the platform in order to further test our custom classifier for Public Opinion of President Donald Trump .

4.1 Collection Methodology
The data collected from Twitter search rapper API as mentioned above. Now a python application is written forth the idea of working over the data. A library is created in order to perform the task with ease. This library works over the data set one by one or as specified, which works provide an efficient use by working over the entire data to be tested. Though, the aspects of user privacy do pose a hindrance over the data, the end result is quite telling . A mechanism is indeed developed to corroborate the whole data as a whole. This involves a NoSQL database where the tweets are saved, and indexed for testing. Python application runs over the NoSQL database in which all user aspects are saved. Now the whole database with its twitter ids, sentiment, data, time, and other fields are stored over a different web server for easy access by other application

V.TESTING THE CLASSIFIER

3800 text statements were used from preliminary collection of Emotion Cuprous and classified into 7 emotions namely anger , disgust , fear , guilt , joy , sadness , shame.

Emotion	Sentences
Anger	When I had been obviously unjustly treated and had no possibility of elucidating this.
Disgust	At a gathering I found myself involuntarily sitting next to two people who expressed opinions that I considered very low and discriminating
Fear	Every time I imagine that someone I love or I could contact a serious illness, even death
Guilt	I feel guilty when when I realize that I consider material things more important than caring for my relatives. I feel very self-centred
Joy	When I pass an examination which I did not think I did well.
Sadness	When I think about the short time that we live and relate it to the periods of my life when I think that I did not use this short time
Shame	When I realized that I was directing the feelings of discontent with myself at my partner and this way was trying to put the blame on him instead of sorting out my own feelings.

Running the classifier like Naïve Bayes , Linear SVM , Logistic Regression and our Custom Consolidated Approach which is a mixture of above specified classifiers which return the Mode of emotion prediction variable based on these algorithms values. Accuracy is as shown in table below.We have used 70% of data for training and 30% of data for testing

Algorithm	Accuracy
Naïve Bayes Algorithm	48.74%
SVM	47.22%
Logistic Regression	45.85%
Consolidated Classifier Approach	48.76%

Table 4 . Classifier and Their Accuracy

```
Most Informative Features
      contains(ashamed) = True          shame : sadnes =     46.5 : 1.0
       contains(guilty) = True          guilt : anger  =     40.4 : 1.0
         contains(fear) = True           fear : anger  =     38.5 : 1.0
       contains(afraid) = True           fear : sadnes =     37.7 : 1.0
        contains(angry) = True          anger : joy    =     36.7 : 1.0
      contains(disgust) = True         disgus : sadnes =     36.1 : 1.0
        contains(died.) = True         sadnes : fear   =     32.3 : 1.0
    contains(disgusted) = True         disgus : fear   =     31.3 : 1.0
         contains(died) = True         sadnes : joy    =     30.0 : 1.0
        contains(drunk) = True         disgus : joy    =     24.2 : 1.0
```

Fig 1 Showing Most Important Features for Naïve Bayes Classifier

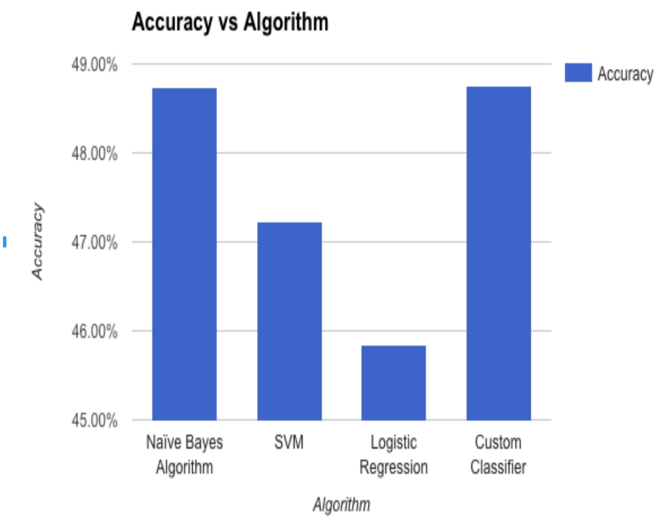


Fig 2 . Graphs Showing Accuracy VS Algorithm

VI.VISUALIZATIONS

In this Section Our Custom Classifier is put to test for real life situations where we apply our custom classifier to the Micro Blogging website like Twitter for Analysis

Tweets Volumes per day

A Line Chart (Fig. 3) shows the amount of tweets for each day over the selected period for the selected keywords

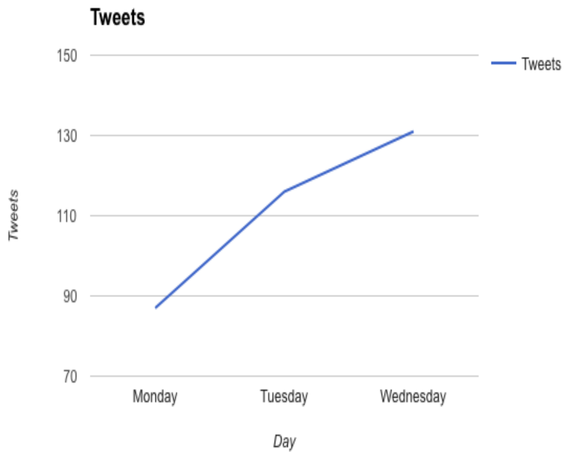
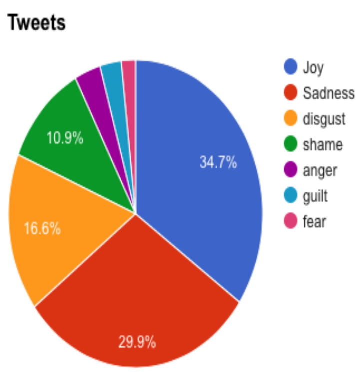


Fig 3

Tweets According to Different Categories

A pie chart (Fig. 4) shows the percentage distribution of tweets in different emotions over a period of 3 days

Emotion	Tweets
Joy	115
Sadness	99
disgust	55
shame	36
anger	11
guilt	9
fear	6



VII. CONCLUSION

The large amount of information contained in microblogging web-sites makes them an attractive source of data for opinion mining and sentiment analysis. In our research, we have demonstrated a method of automatically scraping Twitter data by usage of keywords and geolocation data. We have used three classifiers namely Naïve Bayes , Logistic Regression and SVM to classify the initial data set collected from UCI Machine Learning Repository and then create our Custom Consolidated Approach based on the Information Provided . Our testing also revealed a accuracy of 48.46% for our custom classifier, although the performance and accuracy can be further enhanced through more sophisticated techniques. Lastly we created a web platform to showcase the practical application of the research by applying our custom consolidated approach for Public Opinion on Donald Trump.

As future work, we plan to extend our research to other microblogging platforms such as Weibo, Google+ and Disqus. We also plan to increase accuracy classifying tweets using more sophisticated techniques (*e.g.*(Zhang, Ghosh, Dekhil, Hsu, & Liu, 2011)).

References

[1] Baldwin, B., & Carpenter, B. (2003). Lingpipe. *Available from World Wide Web: http://alias-i.com/lingpipe*.

[2] Barbosa, L., & Feng, J. (2010). Robust sentiment detection on twitter from bi- ased and noisy data. In *Proceedings of the 23rd international conference on computational linguistics: Posters* (pp. 36–44).

[3] Bollen, J., Mao, H., & Pepe, A. (2011). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Icwsm.*

[4] Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1–12.

[5] Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Lrec.*

[6] Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the acl-02 conference on empirical methods in natural language processing-volume 10* (pp. 79–86).

[7] Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., & Liu, B. (2011). Combining lexiconbased and learning-based methods for twitter sentiment

analysis. *HP Laboratories, Technical Report HPL-2011*, *89*

[8] Lillian Lee Bo Pang and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning tech- niques. In *Proceeding EMNLP '02 Proceedings of the ACL- 02 conference on Empirical methods in natural language pro- cessing - Volume 10 Pages 79-86*, pages 81–82, 2008.

[9] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N Project Re- port,Stanford*, pages 1–12, 2009.

[10]BoPangandLillianLee.Opinionminingandsentimentanal- ysis. *Foundations and trends in information retrieval*, 2(1- 2):1–135, 2008.