Sentiment Analysis of Twitter Data using Naïve Bayes with Unigram Approach

P. Bavithra Matharasi[1], Dr. A.Senthilrajan[2]

[1]Lecturer, Dept. of MCA, Mount Carmel College, Bangalore.
[2]Director, Computer Centre, Karaikudi
[1]bavithramatharasip@gmail.com

***Abstract*:** Sentiment analysis has emerged as a widespread and effective technique for opinion mining of web data analysis. The development of the user-generated content has opened new prospects for research in the field of sentiment analysis. This paper analysis a model for sentiment analysis of twitter tweets using Unigram approach of Naïve Bayes. Firstly, tweets need to be downloaded using a free version tool called Node Xl. Once that is done Data pre-processing schemes are applied on the dataset. Secondly, the corpus needs to be trained. Thirdly, the behavior of Naive Bayes is studied in combination with different tweet topics to obtain the results for sentiment analysis.

Keywords: Naïve Bayes classifier, Holdout method, K-fold cross validation, Leave-one-out cross validation

I. INTRODUCTION

Based on Bayes Theorem with hypothesis independent among analyst, the Naïve Bayes classification method comes into picture. In simple language, a Naive Bayes classifier assumes the presence of a particular article in a class is unrelated to the existence of any other feature. Take for example, a fruit may be considered amango if it is yellow, spherical, and has 2.5inches of diameter. Assuming that the features depend alongside each other or the existence of the other additional features, all of these individually contributes to the probability that the fruit is a mango and hence the term used is 'naïve'.

The model of Naïve Bayes is very easy to build and comes very much in handy while using or working with large data sets. Its knows for its simplicity. Being the simplest among the analyzing algorithms, its known to outperform even with highly sophisticated methods of classification It provides a way of classifying and calculating the subsequent probability [14] P (c|x) from P (c), P (x) and P (x|c). Have a look at the equation no.1

$$P\left(\frac{c}{x}\right) = \frac{P(x|c)P(c)}{P(x)} \quad (1)$$

$P\left(\frac{c}{x}\right)$ Posterior Probability
$P(x|c)$ Likelihood
$P(c)$ Class Prior Probability
$P(x)$ Predictor Prior Probability

Above,

- P($c|x$) is the subsequent probability of *class* (c, *target*) given *predictor* (x, *attributes*).
- $P(c)$ is the prior probability of *class*.
- P ($x|c$) is the possibility, which remains the probability of *predictor* given *class*.
- $P(x)$ is the former probability of *predictor*.

Cross Validations

Cross validations, based on the principles of testing the algorithm on a new Dataset yields a better estimate of its performance. The samples used for training are split into validation samples and training samples. The training samples are used to train the algorithm and the validation samples is used as new data to evaluate the performance and working of the algorithm.

Holdout Method

One of the simplest type of validation where the dataset is split into two sets, i.e. the training set and the validation set. The algorithm is trained using the training set only. The algorithm using the validation set then evaluates data.
The evaluation can have a high variance as the evaluation may depend solely on the data that is present in the training set.

K-Fold Cross Validation

K-fold cross validation acts as an improvement of the hideout method. The dataset is repeated k times and divided into k subsets in k-fold cross validation method. In each instance only one of the k sets are used and the remaining k-1 sets are put together to form the training set. The errors across all the trails are then averaged. The disadvantage of K-fold cross validation method is that it takes k times more computational time than because the algorithm is meant to run k times.

Leave-One-Out Cross Validation

The extremely logical forms of k-fold cross validation where k equals the number of data points. The training on the algorithm is done on all data points except for one. It terms to be computationally expensive.

II. LITERATURE SURVEY

It was during the early 1990's that the research was started in the field of sentiment analysis. The term sentiment analysis along with opinion mining was first introduced during the year 2003, during this time the work was very much limited only to subjective detection, sentiment adjectives and interpretation of

metaphors. J.M. Weibe [5] was a research scholar who tried to present an algorithm that was able to identify subjective characters in fictional narrative text based on regularities in the text. M.A.Hearst [6] was another research scholar that had come up with intelligent text based systems to refine the information access task, while J.M. Weibe [5] was undergoing extensive examinations to try to find out if the naturally occurring narratives and regularities with the writings of the authors and come up with an algorithm that would track the point of view on the basis of these regularities.

Another experimental system came into picture, it was called PHOAKS, which was abbreviated as people helping one another to know stuff by L.Terveen [2], this would help users to find the information on the web. This system was known to be using a combined filtering approach to recognize and reuse recommendations.

A browsing method using virtual reviewers for the combined exploration of movie reviews from various viewpoints. Was developed by J.Tatemura. Morinaga et al presented a framework for mining product reputation over the internet, by working in the field of marketing and customer relationship management. This approach that was defined would collect the opinions from the users automatically from the internet and text mining techniques were to obtain the reputation of the product in the market.

An unsupervised method presented by P.D. Turney[1] was used to classify the reviews using a system of thumbs up and down , which would mean thumbs up for recommended and thumbs down for not recommended. It used PMI i.e. point wise mutual Information and document level classification of sentiments to get the average semantic orientation of reviews. The accuracy rate that was obtained was 74%for about 410 reviews. In some time, Turney along with Littman tried to expand their work by presenting an approach that would find the semantic orientation of a text by calculating its statistical association by using a set of positive and negative words using LSA i.e. Latent Semantic Analysis and PMI. This method was tested with 3596 words, which included a combination of 1614 positive words and 1984 negative words and had obtained an accuracy rate of 82%.

Using Standard machine learning techniques a document level sentiment classification was performed by Pang et al[3]. He along with his group mates used Maximum Entropy, naïve bayes and SVM techniques to find results for unigram and bigrams and got a accuracy rate of 82.9% using three fold cross validation for unigrams. The work that they were doing also focused on better understanding of the problems and the difficulties within the sentiment classification task. A classifier was trained using reviews from the major websites by Dave et al[8]. He got a result that showed that the higher order grams can provide better results than unigrams.

M.Rushdi et al [4] discovered the sentiment analysis chore by applying SVM for testing variety of domains of dataset using various weighing schemes. They used three corpora for the experimentation inclusing a new corpus that was introduced by them and performed 10-form as well as 3-ford cross validation for each corpus.

A holistic approach that would infer the semantic orientation of an opinion word that would be based on review context and would combine multiple opinions was proposed by Ding et al[7]. It took into account implicit opinions and handles implicit features that were represented by feature indicators.

Study of sentiments in comparative sentences and web context based sentiments was proposed by Murthy G. and Bing Liu [8]

V Suresh et al [9] presented an approach that used stop words and gaps between stop words as the feature for sentiment analysis.

Algorithm

Given below is the Naïve Bayes classifier. Capital letters depict variables and values are denoted using lower case. Bold chracters are used to depict set of variables.

$\mathbf{X} = \{X1 \ldots Xn\}$	Is a finite set of observed random variables, called features, assumed that each feature takes values from its domain Di.
$\Omega = D1 \times \ldots \times Dn$	the set of all feature
$c \in \{0,\ldots,-1\}$	Unobserved random variable denoting the class of a set of features.
hypothesis $h: \Omega \rightarrow \{0,\ldots,-1\}$	Assigns a class to any agreed set of variables is demarcated as a classifier.
$f(\boldsymbol{x}), c = 0,\ldots,u-1$	class c is assigned a discriminant function
$h(\boldsymbol{x}) = \arg max c \in \{0,\ldots,u-1\} fc(x)$.	The classifier selects the class with the maximum discriminant function on a given set of variables
. $f*(\boldsymbol{x}) = P(C = c \mid X = x)$	set of variables as the discriminant function

Applying Bayes' theorem from
Eq. 1 to this function gives $(C = c \mid \boldsymbol{X} = \boldsymbol{x}) = (\boldsymbol{X}=\boldsymbol{x} \mid C=c)\ P(C=c)\ P(\boldsymbol{X}=\boldsymbol{x})$
Since P(X = x) is the same for all classes it can be ignored. Hence, the Bayes' discriminant function can be written as $f*(\boldsymbol{x}) = P(\boldsymbol{X} = \boldsymbol{x} \mid C = c)\ P(C = c)$, where $P(\boldsymbol{X} = \boldsymbol{x} \mid C = c)\ P(C = c)$ is called the class-conditional probability distribution (CPD).
Thus the Bayes' classifier written as in
Eq. 2 finds the maximum posterior probability hypothesis given x.
$h*(\boldsymbol{x}) = \arg max c P(\boldsymbol{X} = \boldsymbol{x} \mid C = c)\ P(C = c)$ (Eq. 3)
Applying the assumption that features are independent given the class on Eq. 2, we can get the naïve Bayes classifier.
$fc(\boldsymbol{x}) = \prod P(Xj = xj \mid C = c)\ P(C = c)\ nj=1$

III. DATASET

Dataset1: #Achhedin

Achhe din aane waale hain [10] (English translation would mean that Good days are coming) was the Hindi slogan of the indian political party - Bharatiya Janata Party (BJP) during 2014 Indian general parliamentary election. The statement was first used by

BJP's Prime Ministerial candidate Mr. Narendra Modi, assuming that a flourishing future was in store for India if the BJP party would win and come into power. After the BJP's victory in the election, including mentioning the words "*achhe din*" (which means good days) have been used both to express hopefulness and to criticize if anything goes wrong with the Modi government.
Eg. RT @Saloni_shines: Whatt ?? Are petrol and diesel prices hiked again ? No Outrage.. No Media Coverage.. Come on.. #AchheDin
Number of Tweets: 3052 Tweets

Dataset 2: #makeinindia
The Prime Minister, Shri Narendra Modi, today launched the Make in India[11] initiative with an aim to give the Indian economy global recognition. Addressing a gathering consisting of top global CEOs at the event in Vigyan Bhawan in the capital, the Prime Minister said "FDI" should be understood as "First Develop India" along with "Foreign Direct Investment." He urged investors not to look at India merely as a market, but instead see it as an opportunity.
Eg.RT @makeinindia: India records it's highest ever year-on-year FDI inflows. There has never been a better time to #MakeInIndia. https://t.co/…
Number of Tweets: 3096 Tweets

Dataset 3: #pampore
The 2016 Pampore attack[13] was an attack by Lashkar-e-Taiba militants on 25 June 2016, near the town of Pampore in the Indian state of Jammu and Kashmir. **Pampore** or Pampur is a town situated on the eastern side of river Jehlum on Srinagar-Jammu National Highway in Jammu and Kashmir.
Eg.RT @rajnathsingh: Congratulations to Army & Security Forces on successful operation at Pampore.Our forces are extremely capable to counter…
Number of Tweets: 2229 Tweets

Dataset 4: #trumph
Donald John Trump[12] (born June 14, 1946) is an American businessman, television producer, author, politician, and the Republican Party nominee for President
Eg.@cnn well the american should decide on 8 who is better to lead them to promise land. #trumph #clinton
Number of Tweets: 2159 Tweets

Pseudocode
START
Select cross validation method
Add Input file. Read
Cond1- Success Input ? YES/ NO
If YES , Is k-fold cross validation ? YES /NO
Cond1- if NO, Throw error message.
Goto step3
Cond2- If YES, create Training set and test set
Taken data from *C.*
If NO, divide dataset into k sets.
Data from *C.*
Calculate prior probabilities from training set
From *B,* Calculate conditional probabilities for features values in test data
Calculate posterior probabilities for each class
A -Classify . Display the result .
Cond3-is end of test set? YES /NO
If YES, Calculate Accuracy
If NO, goto STEP 11
Display Accuracy
Cond4- If k-fold cross validation ? YES / NO
If YES, Cond5- Is last set? YES / NO
Cond5- If YES, END.
Cond5- If NO, Goto STEP8
Cond4- If NO, END
STOP

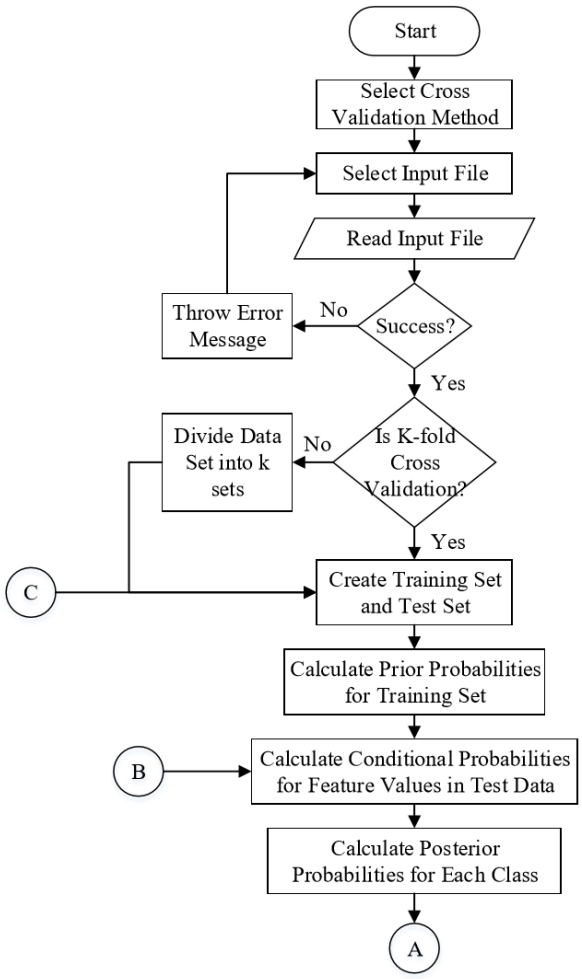

Figure 1: Proposed Framework for Sentiment Analysis using Naïve Bayes

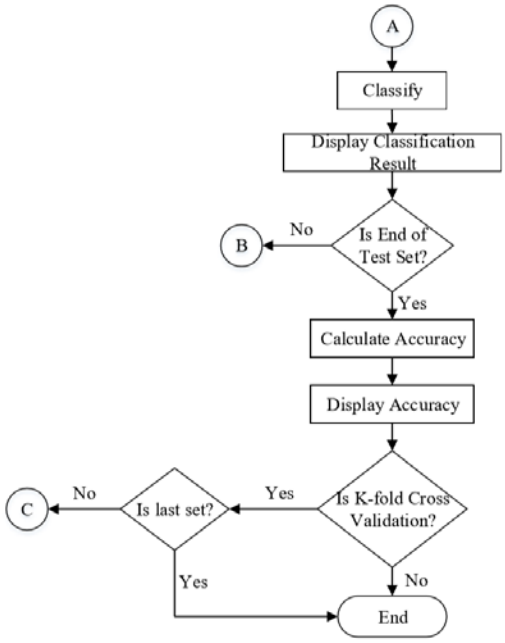

Figure 2: Proposed Framework for Sentiment Analysis using Naïve Bayes

IV. System Design

Created as a desktop application using Microsoft visual Studio along with C# programming language and simple windows form, the naïve bayes classifier is designed for reading a dataset with data that has been categorized.

In order to be read correctly by the program, the following structure needs to be followed.

- The file that is given as input should be a .xls (Microsoft Excel File)or .csv file
- The dataset should first contain the column of tweets and the next column of the tweeted date
- The first column (tweets) can contain any type of data as long as it's the tweet itself.
- The second column should only be of the type date.

The unigram approach is designed in such a way that it can run, three cross validation methods:

1) Holdout method
2) K-fold cross validation
3) Leave-one-out cross validation

Naïve bayes system shown in the unigram approach is designed using windows forms, that reads an input file .xls containing the data set(tweets) according to users choice. Depending on the unigram approach method selected by the user, the input dataset is split into training set and test set.

The training set is then used to calculate the probabilities of each class. The conditional probabilities of each class are calculated using single instance from the test set. Posterior probabilities of each class are then calculated. This process is undergone on each instance of the dataset.

The number of correct classification is obtained which is then used to calculate the accuracy of the naïve Bayes classifier

The workflow of naïve bayes system is shown above in the flowchart (Figure 1 & 2).

V. Results of naïve Bayes

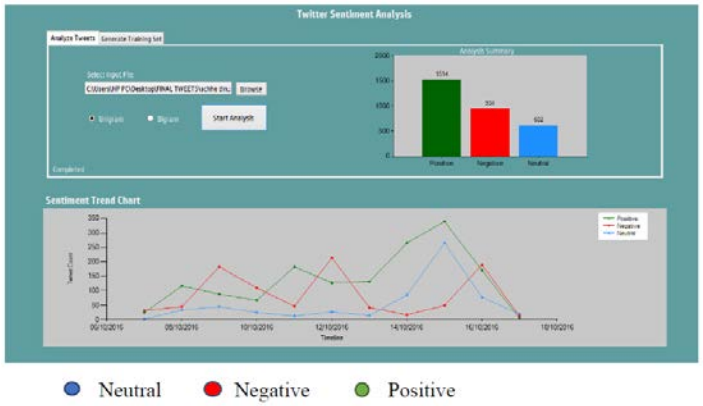

Figure 3: Graphical Results for the dataset1 using Naïve Bayes

In Fig 3 we see the graphical results for the dataset ached in using Naïve Bayes. The positive tweets were 49% for unigrams using naïve . On addition, The negative tweets were found to be 30% and the neutral tweets were found to be 20%. It is concluded that positive tweets prevail over the other two tweet rates.

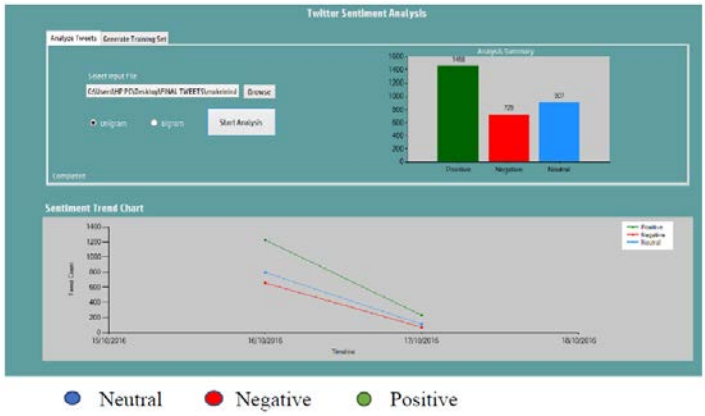

Figure 4: Graphical Results for the dataset2 using Naïve Bayes

Figure 5: Graphical Results for the dataset3 using Naïve Bayes

In Fig 5 we see the graphical results for the dataset trumph using Naïve Bayes. The positive tweets were 54% for unigrams using naïve. On addition The negative tweets were found to be 23% and the neutral tweets were found to be 21%. It is concluded that positive tweets prevail over the other two tweet rates and the difference between neutral and negative was found to be only 2%.

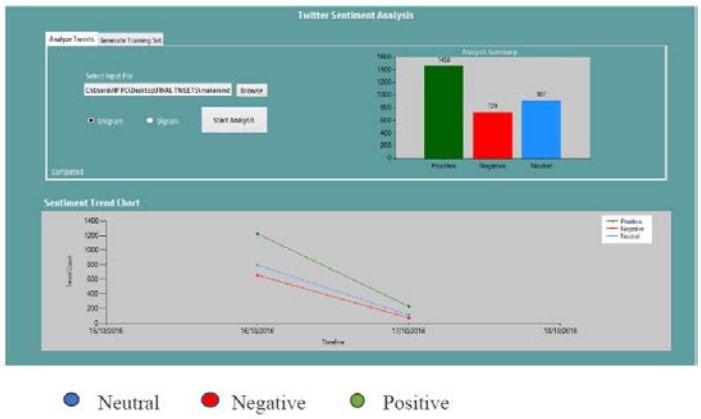

Figure 6: Graphical Results for the dataset4 using Naïve Bayes

In Fig 6 we see the graphical results for the dataset make in India using Naïve Bayes. The positive tweets were 47% for unigrams using naïve. On addition The negative tweets were found to be 23% and the neutral tweets were found to be 29%. It is concluded that positive tweets prevail over the other two tweet rates.

VI. Conclusion

In this paper, we designed and developed a naïve Bayes classifier that was designed to read any data set with categorical data and a prescribed structure in the input excel file. The classifier was tested using four different data sets generated from node XL software using twitter API. The naïve Bayes methods were used to calculate the accuracy of the classifier.

For all the data set, the positive, negative and neutral tweets were found. Also a time based analysis was done using line graph to check the day wise rating of the tweets.

It can be observed from this paper, that the results of the naïve Bayes classifier showed some results that were in general and with a lot of errors that were not been able to be handled by the Bayes algorithm.

References

[1] P.D. Turney, "Unsupervised Learning of Semantic Orientation from a Hundred-Billion", May 16, 2002.
[2] Loren Terveen, Will Hill, Brian Amento, David Mc Donald and Josh Creter "PHOAKS: a system for sharing recommendations", March 1997
[3] B. Pang et al, "sentiment classification using machine learning techniques" 2002,
[4] M. Rushdi Saleh et al, " SVM to classify opinions in different domains", 2011
[5] WiebeJanyce " Identifying subjective characters in narrative, Proceedings of the International Conference on Computational Linguistics (COLING-1990)." , 1990
[6] Hearst M., 1992, Direction-based text interpretation as an information access refinement in TextBased Intelligent Systems, P. Jacobs, Editor 1992, Lawrence Erlbaum Associates, 257-274.
[7]Xiaowen Ding et al, 2008, A holistic lexicon-based approach to opinion mining, WSDM"08, February 11-12, 2008, Palo Alto, California, USA.
[8] Murthy G. and Bing Liu, 2008, Mining opinions in comparative sentences, Proceedings of the 22nd international conference on computational linguistics (Coling 2008), Manchester, August 2008, 241248.
[9] V. Suresh et al, 2011, A Non-syntactic Approach for Text Sentiment Classification with Stopwords, WWW 2011,March 28–April 1, 2011,Hyderabad, India
[10] en.wikipedia.org/wiki/Achhe_din_aane_waale_hain
[11] en.wikipedia.org/wiki/Make_in_India
[12] en.wikipedia.org/wiki/Donald_Trump
[13] en.wikipedia.org/wiki/2016_Pampore_attack
[14] analyticsvidhya.com/blog/2015/09/naive-bayes-explained/