Design and Implementation Sentimental Text Classification Using Machine Learning

Sarika Rai[1], Awdhesh kumar[2]
[1, 2] Department of CS
[1, 2] LNCTS, Indore(M.P.)

Abstract- *The key aim of the proposed work is to find a data mining based technique for classifying the text according to the text sentiments are achieved successfully. The implementation of semantics based text analysis classifier is performed successfully and their performance is also estimated. According to the obtained performance outcomes the system, works accurately and efficiently as compared to traditional system but the performance is not much acceptable due to high time complexity. In near that is required to introduce more literature and effort to make a less complex system for improving the current issues of the computational complexity. After implementation of the system the performance of the system in terms of accuracy, error rate, space complexity and time complexity is estimated and compared with a traditional classifier namely SVM (support vector machine).*

Keywords- SVM, Neural Network , Sentiments Analysis

I. INTRODUCTION

The data mining and machine learning provides the technique to analyze a huge amount of data using computer applications. These applications first trained on previously specified patterns and during their analysis similar patterns are results on which these algorithms are trained. Thus if an algorithm is trained to find a sequential data then during the testing or real time data analysis the algorithm only returns the sequential data patterns. The learning of the algorithms are also depends on the kind of data on which the training is made if the data is well defined in a given set of labels then that is known as supervised learning and if the data is not described in a predefined class labels than that is known as cluster analysis or the unsupervised learning.

In this presented work the supervised learning is utilized to train the algorithm for identifying there class labels. These class labels are distributed in two classes sad and happy according to their semantics. In order to identify their patterns or semantics, the probability theory and machine learning algorithm making combined efforts. Therefore two different classifiers are required to organize in a same place. The Bayesian classifier is first employed to find the occurrence of a word in a given text for sad data representation and also the probability is measured for happy words probability of occurrence. These estimated probabilities are utilized with the machine learning algorithm namely neural network to train them.

In addition of that some additional efforts are also required to accurate classification of data, basically in this experimentation text data is analyzed for classification. The text data classification having it's own issues and limitations such as huge amount of data, noise in data and unstructured manner of arrangement. Therefore a strong pre-processing technique is also need to associate with the given text processing technique. therefore the entire system process involve the three major steps first the pre-processing of data, preparing learning of data model and finally the data testing on data.

This section describes the basic overview of the presented system in further section the detailed model-ling of the proposed system is given where the system flow and the process of all the training data and testing of the training patterns are described in detail.

The domain of sentiment analysis handles the analysis of sentiments found in text documents. A basic task is sentiment classification, where a definite amount of text is sorted into classes which relate to, e.g. positivity or negativity of expressed ideas. This is an application of dimensionality reduction techniques for two sentiment classification issues:

(1) Document polarity classification, where documents representing complete reviews are classified into positive or negative.

(2) Sentence polarity classification, which deals with polarity classification of individual sentences [6].

II. RESEARCH ELOBRATION

Qiang Yan et al [1] proposed a social network based human dynamics model in this paper, and pointed out that inducing drive and spontaneous drive lead to the behaviour of posting microblogs.

Unfortunately, collecting relevant data can be costly and finding meaningful information for analysis is challenging. A growing number of Location-based Social Network services provide time-stamped, geo-located data that opens new opportunities and solutions to a wide range of challenges. Such spatiotemporal data has substantial potential to increase situational awareness of local events and improve both planning and investigation. However, the large volume of unstructured social media data hinders exploration and examination. To analyze such social media data, the system provides the analysts with an interactive visual spatiotemporal analysis and spatial decision support environment that assists in evacuation planning and disaster management. ***JunghoonChae et al [2]*** demonstrate how to improve investigation by analyzing the extracted public behaviour responses from social media before, during and after natural disasters, such as hurricanes and tornadoes.

Microblogging services such as Twitter are said to have the potential for increasing political participation. Given the feature of "retweeting" as a simple yet powerful mechanism for information diffusion, Twitter is an ideal platform for users to spread not only information in general but also political opinions through their networks as Twitter may also be used to publicly agree with, as well as to reinforce, someone's political opinions or thoughts. Besides their content and intended use, Twitter messages ("tweets") also often convey pertinent information about their author's sentiment. In this paper, ***Stefan Stieglitz et al [3]*** seek to examine whether sentiment occurring in politically relevant tweets has an effect on their retweet ability (i.e., how often these tweets will be retweeted). Based on a data set of64,431 political tweets, they find a positive relationship between the quantity of words indicating affective dimensions, including positive and negative emotions associated with certain political parties or politicians, in a tweet and its retweet rate. Furthermore, they investigate how political discussions take place in the Twitter network during periods of political elections with a focus on the most active and most influential users. Finally, authors conclude by discussing the implications of results.

Using semantic technologies for mining and intelligent information access to microblogs is a challenging, emerging research area. Unlike carefully authored news text and other longer content, tweets pose a number of new challenges, due to their short, noisy, context dependent, and dynamic nature. Semantic annotation of tweets is typically performed in a pipeline, comprising successive stages of language identification, tokenisation, part-of-speech tagging, name identity recognition and entity disambiguation (e.g. with respect to DBpedia). Consequently, errors are cumulative, and earlier-stage problems can severely reduce the performance of final stages. ***Leon Derczynski et al [4]*** present a characterization of genre-specific problems at each semantic annotation stage and the impact on subsequent stages. Critically, they evaluate impact on two high-level semantic annotation tasks: named entity detection and disambiguation. Results demonstrate the importance of making approaches specific to the genre, and indicate a diminishing returns effect that reduces the effectiveness of complex text normalization.

III. PROPOSED SYSTEM ARCHITECTURE

The proposed system is a text classification data model.That classifies text according to their semantics in two classes. Therefore the presented classifier is a binary classifier which is works in two stages training and testing. In training phase the training data samples are used for perform learning on the data patterns and then after the trained model is used to classify the data according to their previous training trends. In order to explain the entire system processes the figure 1 contains the system architecture and their sub-components are described as the following.

(A) Input Data set:

The system needs to learn with the semantics text therefore the large text can affect the performance of classifier. Therefore in order to train and accurately classify data the input training data is taken from the twitter dataset which is frequently used for sentiment analysis of text. That dataset organized in two columns first the training text patterns and second the class labels. The given class labels are having only two values namely 0 and 1.

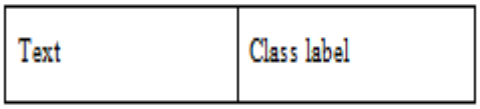

Figure1: Training samples

In formal data mining techniques the pre-processing leads to cleaning operations that enhances the quality of training patterns to adopt the classes and patterns to make easier training and perform more effective training. In this given pre-processing the data is refined for finding the more appropriate terms that are actually belongs to the classification. Therefore the stop words that are frequently arrived on text are removed first form entire set of data. After that the refined text and their class labels are stored in a database.

(B) Bayesian classifier

The Naive Bayes classification algorithmic rule is a probabilistic classifier. It is based on probability models that incorporate robust independence assumptions. The independence assumptions usually don't have an effect on reality. So they're thought of as naive. You can derive probability models by using Bayes' theorem (proposed by Thomas Bayes). Based on the nature of the probability model, you'll train the Naive Bayes algorithm program in a very supervised learning setting. In straightforward terms, a naive Bayes classifier assumes that the value of a specific feature is unrelated to the presence or absence of the other feature, given the category variable. There are two types of probability as follows.

- Posterior Probability [P (H/X)]
- Prior Probability [P (H)]

Where, X is data tuple and H is some hypothesis. According to Baye's Theorem

$$P\left(\frac{H}{X}\right)=\frac{P\left(\frac{X}{H}\right)P(H)}{P(X)}$$

Using this algorithm the probability of each word is estimated for become in happiness text or others. To understand the probability estimation of each word the table 3.2 can helps.

Word	For happiness	For sadness
Wow	0.352	.037

Table 2 bay's probability of words

According to the given table each word's probability is measured and their training is made using the neural network. The next section shows the neural network and their training algorithm.

(C) Neural network

The implementation of neural network is defined in two phases' first training and second prediction: training method utilizes data and designs the data model. By this data model next phase prediction of values is performed.

Training:

1. Prepare two arrays, one is input and hidden unit and the second is output unit.
2. Here first is a two dimensional array is used and output is a one dimensional array Y_i.
3. Original weights are random values put inside the arrays after that the output is given as**.**

$$x_j=\sum_{i=0} y_i W_{ij}$$

Where, y_i is the activity level of the j^{th} unit in the previous layer and is the weightof the connection between the i^{th} and the j^{th} unit.

4. Next, action level of y_i is estimated by sigmoidal function of the total weighted input.

$$y_i=\left[\frac{e^x \quad e^{\ x}}{e^x+e^{\ x}}\right]$$

When event of the all output units have been determined, the network calculates the error (E) given in equation**.**

$$E=\frac{1}{2}\sum_i (y_i \quad d_i)^2$$

Where, y_i is the event level of the j^{th} unit in the top layer and d_i is the preferred output of the j_i unit.

Calculation of error for the back propagation algorithm is as follows:

- Error Derivative (is the modification among the real and desired target:

$$EA_j=\frac{\partial E}{\partial y_j}=y_j \quad d_j$$

- Error Variations is total input received by an output changed:

$$EI_j=\frac{\partial E}{\partial X_j}=\frac{\partial E}{\partial y_j}X\frac{dy_j}{dx_j}=EA_j y_j(1-y_i)$$

- In Error Fluctuations calculation connection into output unit is required:

$$EW_{ij}=\frac{\partial E}{\partial W_{ij}}=\frac{\partial E}{\partial X_j}=\frac{\partial X_j}{\partial W_{ij}}=EI_j y_i$$

- Overall Influence of the error:

$$EA_i=\frac{\partial E}{\partial y_i}=\sum_j \frac{\partial E}{\partial x_j}X\frac{\partial x_j}{\partial y_i}=\sum_j EI_j W_{ij}$$

The neural network process discussed now how the training is made for the neural network to learn the training pattern is discussed. Therefore for each training sample is first prepared for performing the training therefore for each sample an aggregate probability is computed and then with their class labels the training is performed. For example the training for a text string is reported using table 3.

Text	For happiness	For sadness	Class
Hello friend how are you	0.731	.3473	0

Table 3 sample training data

Here the probability of text for happiness and sadness is given as input to neural network and the desired outcome is produced as their defined class label 0 or 1. After training for all the data samples which is provided as input training sample the testing on a separately prepared set is performed and their performance is measured. The given chapter provides the understanding of the proposed classifier algorithm. Therefore first the concept behind the algorithm development is defined and then after the required algorithms is listed. After that using defined algorithm the proposed algorithm is formulated and described in detail. Finally using the flow chart the flow of data during the algorithm process is described. This chapter provides a detailed understanding about the proposed concept of sentiments based text analysis and the next chapter reports the implementation of the work.

IV. RESULT ANALYSIS

After implementation of the proposed system the performance of proposed classification technique and previously available technique is evaluated and compared using their performance graphs. The given chapter provides the detailed discussion about the preformed experiments and their results.

1. Accuracy

In a data mining based classification system the amount of correctly recognized patterns are known as the classification accuracy. The accuracy of the system in terms of percentage can be computed using the following formula.

$$accuracy = \frac{accurately\ classified\ patterns}{total\ input\ patterns} X100$$

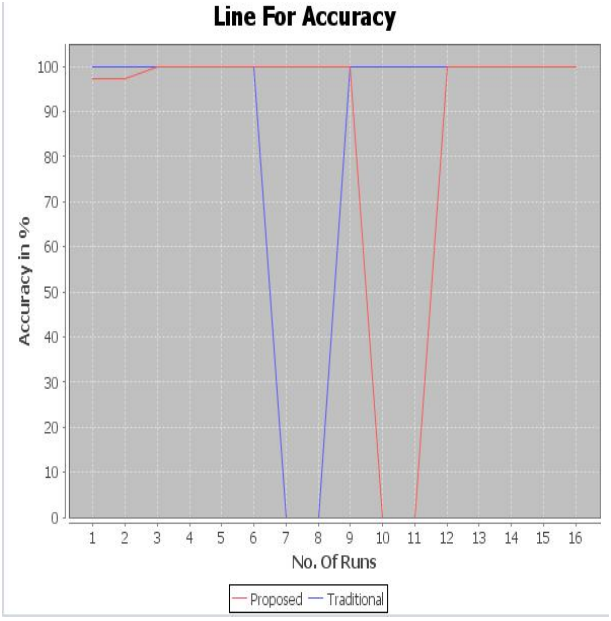

Figure 1 : Accuracy

The values of graph is represented using where the amount of accuracy of the proposed algorithm is given in first column and the second column contains the values of traditional approach namely SVM. In the similar ways the given graph as given contains the comparative accuracy of both the algorithms. In this figure blue line shows the proposed algorithms' performance and the red line shows the performance of the traditional approach. For demonstrating the performance of the system X axis contains the amount of data during the training and testing and Y axis contains the obtained performance in terms of accuracy. According to the obtained results the performance of the proposed classification technique provides more accurate results as compared to the traditional approach.

2. Error rate

The amount of data misclassified during classification of algorithms is known as error rate of the system. That can also be computed using the following formula.

$$error\ rate = \frac{total\ misclassified\ patterns}{tal\ input\ patterns} X100$$

Or

$$error\ rate = 100 - accuracy$$

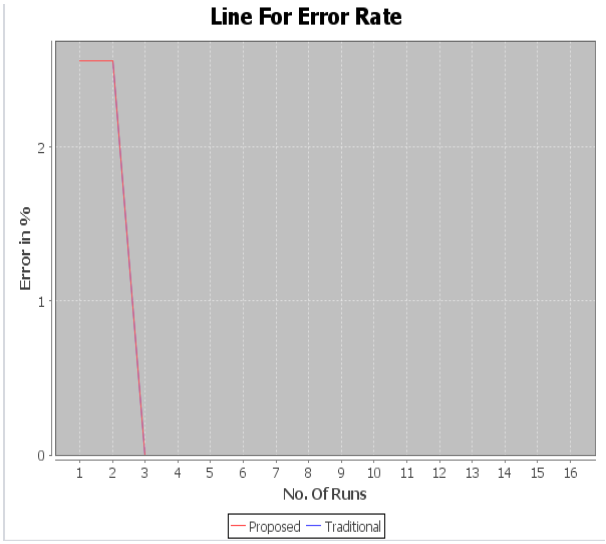

Figure 2 : Error

shows the comparative error rate of both the implemented algorithms namely SVM and proposed hybrid classifier. In order to show the performance of the system the X axis contains the amount of data used for training and the Y axis shows the performance in terms of error rate percentage. The performance of the proposed classification is effective and efficient during different experimentations and reducing with the amount of data increases. Thus the presented classifier is more efficient and accurate than the traditional approach of text classification.

3. Memory uses

Memory consumption of the system also termed as the space complexity in terms of algorithm performance. That can be calculated using the following formula:

$$memory\ consumption = total\ memory - free\ memory$$

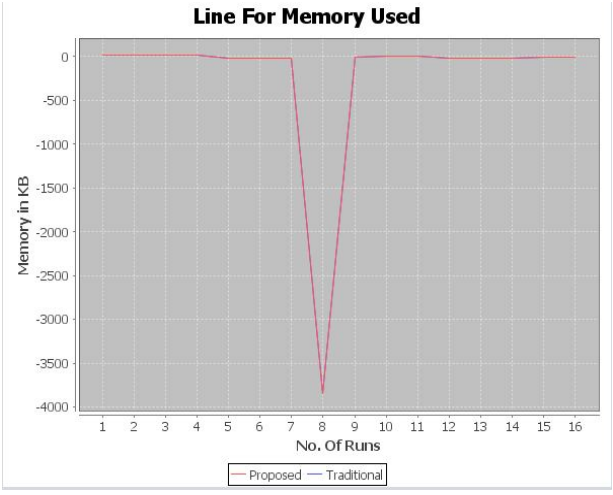

Figure 3 : Memory uses

The amount of memory consumption depends on the amount of data reside in the main memory, therefore that affect the computational cost of an algorithm execution. The performance of both the implemented classifiers for sentiment classification is given using . in this diagram the blue line shows the performance of the proposed classifier and the red line shows the performance of traditional classification scheme. For reporting the performance the X axis of figure contains the amount of data required to execute using the algorithms and the Y axis shows the respective memory consumption during experimentations. According to the obtained results the performance of both the algorithm demonstrate similar behaviour with increasing size of data, but the proposed technique consumes additional memory as compared to the traditional algorithm because the traditional algorithm is implemented with the single algorithm implementation and proposed algorithm requires to execute two different classification techniques. The implementation of the proposed concept is provided using the JAVA based IDE namely NetBeans. After implementation of the system the performance of the system in terms of accuracy, error rate, space complexity and time complexity is estimated and compared with a traditional classifier namely SVM (support vector machine). The comparative performances of both the algorithms are summarized in the table.

S. No.	Parameters	Proposed algorithm	Traditional algorithm
1	Accuracy	High	Low
2	Error rate	Low	High
3	Memory consumption	Low	High
4	Time consumption	High	Low

Table 4 performance summary

According to the comparative performance study the proposed classification technique is performed efficiently and accurately due to high classification rate and improving error rate during prediction. Additionally the space complexity of the proposed system is less but the performance of the system leaked in terms of time complexity. That required more time for accurate learning.

V. CONCLUSION

The presented work is dedicated to finding an efficient and accurate approach by which the unstructured data and their semantics can be analyzed in a more effective way. In order to obtain such kind of method a number of techniques are analyzed and a most promising effort is obtained. Using this approach a new hybrid classifier is prepared for improving the performance of existing classification technique. The proposed technique, concept hybridizes the concept of Bayesian classification and after that the data model is prepared using the back-propagation neural network. In a first step the twitter's labeled data is used by the Bayesian classifier and the probability of each token in both the class classification is placed and then after the computed probability is converted into weights. These weights are distributed in both the defined classes and further used for neural network training.

During the classification test cases the data are again analyzed for with the same process and weights are aggregated for defining the text class labels. According to this test cases the results and performance of the system are estimated and summarized.

REFERENCES

[1] Qiang Yan, Lianren Wu, LanZheng, "Social network based microblog user behavior analysis", 2012 Elsevier B.V. All rights reserved

[2] JunghoonChae, Dennis Thom, Yun Jang, SungYe Kim, Thomas Ertl, David S. Ebert, "Public behavior response analysis in disaster events utilizing visualanalytics of microblog data", & 2013 Elsevier Ltd. All rights reserved

[3] Stefan Stieglitz, Linh Dang-Xuan, "Political Communication and Influence through Microblogging 6 An Empirical Analysis of Sentiment in Twitter Messages and Retweet Behavior", 2012 45th Hawaii International Conference on System Sciences

[4] Leon Derczynski, Diana Maynard, NirajAswani and KalinaBontcheva, "Microblog-Genre Noise and Impact on Semantic Annotation Accuracy", 24th ACM Conference on Hypertext and Social Media1–3 May 2013, Paris, FranceCopyright 2013 ACM

[5] Luiz F. S. Coletta, N´adia F. F. da Silva, Eduardo R. Hruschka,Estevam R. Hruschka Jr., "Combining Classification and Clustering for TweetSentiment Analysis", 2014 Brazilian Conference on Intelligent Systems (BRACIS), 18-22 Oct. 2014

[6] Umajancy. S, Dr. Antony SelvadossThanamani, "An Analysis on Text Mining –Text Retrieval and Text Extraction", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 8, August 2013

[7] MilošRadovanović, MirjanaIvanović, "Text Mining: Approaches And Applications", Abstract Methods and Applications in Computer Science (no. 144017A), Novi Sad, Serbia, Vol. 38, No. 3, 2008, 227-234

[8] Vishal Gupta, Gurpreet S. Lehal, "A Survey of Text Mining Techniques and Applications", Journal of Emerging Technologies in Web Intelligence, Vol. 1, No. 1, August 2009

[9] P. Bhargavi, B. Jyothi, S. Jyothi, K. Sekar, "Knowledge Extraction Using Rule Based Decision Tree Approach", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.7, July 2008

[10] Xia Hu, Lei Tang, Jiliang Tang, Huan Liu, "Exploiting Social Relations for Sentiment Analysisin Microblogging", WSDM '13, February 4–8, 2013, Rome, Italy, Copyright 2013 ACM 978-