International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 3, Issue 4 July-August 2014 **ISSN 2278-6856**

Analyzing Sentiment of Movie Review Data using Naive Bayes Neural Classifier

Lina L. Dhande[1] and Dr. Prof. Girish K. Patnaik[2]
[1,2] Department of Computer Engineering,
SSBT's College of Engineering and Technology, Bambhori, Jalgaon, Maharashtra, India,

Abstract

Now a days sentiment analysis is active field of research, to extract people's opinion about particular product or service. The most useful application of sentiment analysis is the sentiment classification of product reviews. The task of sentiment classification is to classify reviews of user as positive or negative from textual information alone. For that purpose many researchers used data mining classification techniques such as Naive Bayes classifier with strong independence assumption. But Naive Bayes classifier lack in accuracy for many complex real-world situations where there exists dependency among features. Further, the Neural Network with appropriate network structure is good enough to handle the correlation or dependence between input variables. In proposed system, the Naive Bayes and Neural Network classifier are combined for sentiment classification. In Experimental results, the movie review is classified into positive or negative polarities of sentiment using classifiers. The accuracy of sentiment analysis is increased upto 80.65% by combining Naive Bayes classifier with Neural Network for unigram feature on movie review dataset.

Keywords: Sentiment Analysis, Sentiment Classification, Naive Bayes, Neural Network Classifier.

1. INTRODUCTION

Language is a powerful tool to communicate and convey information. It is also a means to express emotion and sentiment. Sentiment analysis is the field of study that analyzes people's opinions, sentiments, evaluations, attitudes and emotions towards entities such as products, services, organizations, individuals, issues, events, movies and topics. In simple words, it is used to track the mood of the public. It uses natural language processing and data mining techniques to the problem of extracting opinions from text. In data mining research field, machine learning techniques have been applied to automatically identify the information content in text. In recent years, the exponential increase in the Internet usage and exchange of public opinion is the driving force behind sentiment analysis. The web is a huge repository of structured and unstructured data. The analysis of this data to extract latent public opinion and sentiment is a challenging task. Generally speaking, sentiment analysis aims to determine the attitude of a speaker or a writer with respect to the overall document. Sentiment analysis is performed at different levels of granularity with different levels [1]. The most useful application of sentiment analysis is the sentiment classification of product reviews. This sentiment classification can be categorized into positive and negative. Positive rating returns are good, negative returns are bad review. Users express opinions through review sites such as Amazon, Internet Movie Database (IMDB) and Epinions, as well as through blogs, discussion forums, peer-to-peer networks, user feedback, comments and various types of social network sites. These kinds of online media have resulted in large quantities of textual data containing opinion and facts. Over the years, there has been extensive research aimed at analyzing and classifying text and data, where the objective is to assign predefined category labels to documents based upon learned models. This has led to the development of sentiment analysis and classification systems. Sentiment analysis and classification [2] [3] are technically challenging. Bo Pang et al. [4] used movie reviews to train an algorithm that detects sentiment in text. Movie reviews are a good source for this kind of work because authors clearly express an opinion and authors are accompanied by rating that makes it easier to train learning algorithms on this data. Most text classification methods that classify a given document into one of the predefined classes are based on bag of words [5], where each document is represented by set of words.

The rapid growth of the World Wide Web has facilitated increased online communication and opened up newer avenues for the general public to post their opinions online. This has led to generation of large amounts of online content rich in user opinions, sentiments, emotions, and evaluations. Recently, many web sites have offered reviews of items like books, cars, snow tires, vacation destinations, movies etc. Informers describe the items in some detail and evaluate them as good/bad, preferred/not preferred and positive/negative. That is, whether people recommend or do not recommend a particular item. For example, people express their views for movies as, "I like movie and the story is fantasying". In sentimental classification problem, movie review mining is a challenge. The inspiration for this work has come from studies in classification. Research area of sentiment analysis is motivated for doing sentiment classification using new combination of classifier for improving accuracy. A modern approach towards sentiment classification is to use machine learning techniques which inductively build a classification model of a given set of categories by training several sets of labeled document. Popular machine learning methods include Naive Bayes, K-Nearest Neighbor, Support Vector Machines and Neural Network. Among them,

Naive Bayes classifier is more appropriate to be extended. Many researchers have done sentiment classification by using Naive Bayes classifier. But Naive Bayes classifier has major limitation that the real-world data may not always satisfy the independence assumption among attributes. Hence, it affects the accuracy of Naive Bayes classifier. So combine other method with Naive Bayes for sentiment analysis. Objective of this work is to effectively classify movie review in positive and negative polarities and to increase the accuracy of sentiment analysis. In data mining, Naive Bayes and Neural Network classifier is used for classification task. Naive Bayes classifier is simple probabilistic classifier. It is a technique that applies to a certain class of problems, namely those that can be phrased as associating an object with a discrete category. The performance of Naive Bayes is poor when features are co-related to each other. The Neural Network is good enough to handle the correlation and dependence between input variables. So by combining Naive Bayes classifier with Neural Network will increase the accuracy of sentiment classification in real world dataset. Naive Bayes Neural classifier is proposed to classify the standard movie dataset. The paper proceeds as follows. Section 2 presents the related work of sentiment analysis. Naive Bayes classifier, Neural Network classifier and Naive Bayes Neural classifier is presented in Section 3. Section 4 presents the experimental setup of proposed system with result. Discussion is discussed in Section 5. Finally conclude the paper and future work.

2. Related Work

Much work has recently been undertaken in sentiment analysis over the last few years. Pang and Lee [4] gives an excellent review. Work has been done specifically on sentiment analysis and even more recently work has been carried out on mining tweets from Twitter. DENG et al. (2011) uses sentiment analysis for Stock Price Prediction in [6]. The classifier tries to classify the review into positive or negative category. The classification result will be the basis of the rating. With the proportion of positive and negative reviews, the system could provide the rating information to end users. Bo Pang et al., in [4], presented sentiment classification using machine learning techniques. For the effectiveness of classification of documents by overall sentiment used learning methods Naive Bayes, Maximum Entropy classification and Support Vector Machines. The unigrams and bigrams features were used for classification. The movie-review corpus with randomly selected positive sentiment and negative sentiment reviews were used for experimental setup. Result of the machine learning algorithms clearly surpasses the random-choice baseline of 50%. Authors also handily beat two human-selected-unigram baselines of 58% and 64% and performed well in comparison to the 69% baseline achieved via limited access to the test-data statistics. Whereas the accuracy achieved in sentiment classification is much lower when compared to topic based categorization. Authors proposed system which effectively classifies movie reviews in polarities but it require more training time for all of three classifiers. Yaying Qiu et al., in [7], constructed extend bayes model with assigning weights to important features. Authors have researched problems in how to classify Chinese text efficiently and effectively. For that purposed authors used the approaches with Naive Bayes and CF methods are used to measure the relevance between a feature and a category to make up the deficiency of CHI-Square statistic method. Authors select best features based on a proposed method called CHCFW to reinforce the distribution of key features in the document and remove the disturbed features. Experiment result had shown that how the size of best feature set by chosen influence the accuracy using the CHCFW method and the ratio of training set is 80%. Authors effectively classify Chinese text but some problems needs to be improved that is calculate the weight of each feature which needs a relatively long time, therefore, the whole process is somehow time consuming. Duan et al., in [8], presented mining online user reviews for both quantitative aspects and textual content from multi-dimensional perspectives. In recent years, online user generated content exploded which revolutionizing the hotel industry. Online user generated reviews for the hotel industry used as data source. Experiment had shown the results of sentiment analysis with increasing accuracy in measuring and capturing service quality dimensions. That was compared with existing text mining studies. After used econometric modeling technique to examine the potential differential effect of different service quality dimensions. Tao Xu et al., in [9], used online forum for sentiment analysis. The online forum like BBS provides a communication platform for people to discuss and express their views. Back-Propagation Neural Network (BPNN) used for extracted feature and Vector Space Model (VSM) used for represented text documents as vectors which is an algebraic model. Experiment had shown three layer Neural Networks to predict hotness of a topic. The Neural Network consisted the input layer has four neurons, hidden layer has eight neurons and output has one neuron. Approach is divided into data collected and preprocessing, Neural Network training and prediction. SINA reading forum is used for collected data. Experiment effectively predicts the hotness of topics. Pablo Gamallo et al., in [10], proposed sentiment analysis on Spanish tweets. Authors focused on stressed the microblogging service Twitter. As Twitter can be seen as a large source of short texts (tweets) containing user opinions. The task of making sentiment analysis from tweets is a hard challenge. Authors used approach, Naive-Bayes classifier for detecting the polarity of Spanish tweets. Experimental results shown a performance of the system about 67% accuracy which used to classify sentiment analysis for detect six sentiment categories.

Mohamed Abdel Fattah, in [11], sentimentally classifies movie reviews. Authors used Gaussian Mixture Model (GMM) and Feed Forward Neural Network (FFNN) for sentiment classification. GMM has been exploited since it has often been found to provide good classification results. The achieved result using GMM is 81.5% based on accuracy measurement. A set of highest score sentences are chronologically specified as a document summary based on 30% compression rate by using feed forward neural network. Much of the work is done in sentiment analysis using Naive Bayes classifier [4], [7], [12] and [13]. Many researchers used Naive Bayes classifier to analyze sentiment over movie review dataset. Problem with Naive Bayes classifier is always assume conditional independence that clearly does not hold real-world situation. Literature survey seen that there is not work done in sentiment analysis of movie review using Neural Network. Neural Network is universal approximater for correlated data.

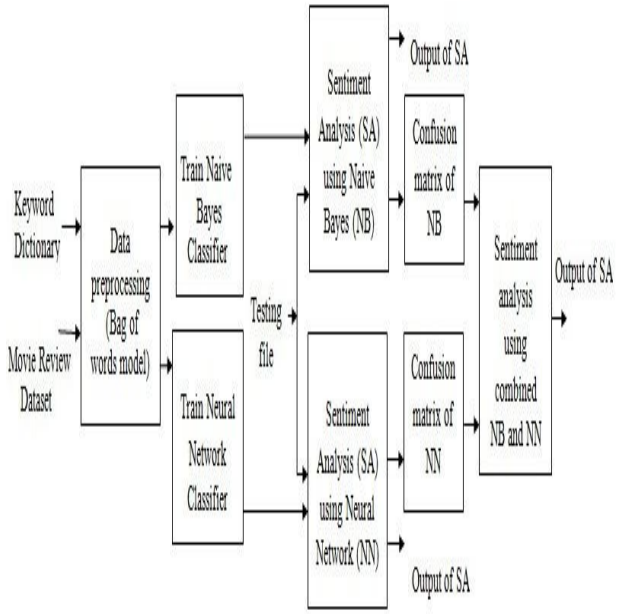

Figure 1: Proposed System

3. METHODOLOGY

A modern approach towards sentiment classification is to use machine learning techniques which inductively build a classification model of a given set of categories by training several sets of labeled document. Popular data mining methods include Naive Bayes, K-Nearest Neighbour, Support Vector Machines and Neural Network. In proposed system, Naive Bayes and Neural Network supervised methods are used for classification.

3.1 Architecture

The proposed system architecture is shown in Figure 1. Input to proposed system is the Movie review dataset [14] and keyword dictionary. The WordStat Dictionary [15], is the commonly used dictionary in sentiment analysis. WordStat dictionary contains words according to multiple categories, including positive or negative. This dictionary includes 354 positive words and 2396 negative words. Data reprocessing is done by Bag of Words (BOW) model. In BOW model, the occurrence of each word is used as a feature for training a classifier. It is also used for unigram feature for classification. A unigram feature marks the presence or absence of a single word within a text. The processed data is passed to Naive Bayes classifier and Neural Network classifier for classification. Testing file is the one of the input of sentiment analysis. Trained Naive Bayes classifier and Neural Network classifier is a second input given to both sentiment analysis blocks. Sentiment analysis using Naive Bayes and Neural Network classifier is produced the output and evaluates confusion matrix. A confusion matrix contains information about actual and predicted classifications done by classifier. After that depending on confusion matrix size, sentiment analysis using combined Naive Bayes (NB) and Neural Network (NN) approach gives correct result of classification.

3.2 Naive Bayes Classifier

Naive Bayes classifier is a simple model for classification [16]. It is simple and works well on text classification. It is a probabilistic classifier based on applying Bayes' theorem with strong independence assumptions. This is the simplest form of Bayesian Network, in which all attributes are independent given the value of the class variable. This is called conditional independence. It assumes each feature is conditional independent to other features given the class. A Naive Bayes classifier is a technique that applies to a certain class of problems, namely those that phrased as associating an object with a discrete category. From numerical based approach group, Naive Bayes has several advantages such as simple, fast and high accuracy. In K. Ming Leung [17] describes the Bayes rule.

$$\gamma(\alpha \mid \beta) = \frac{\gamma(\alpha) * \gamma(\beta \mid \alpha)}{\gamma(\beta)}$$

Where α : Specific class

β : Document wants to classify

$\gamma(\alpha)$ and $\gamma(\beta)$: Prior probabilities

$\gamma(\alpha \mid \beta)$ and $\gamma(\beta \mid \alpha)$: Posterior probabilities

The value of class α might be positive or negative. Document is a review of particular movie. The multinomial model [18] of Naive Bayes captures word frequency information in documents. The Maximum Likelihood Estimate (MLE) is simply the relative frequency and corresponds to the most likely value of each parameter given the training data. For the prior probability this estimate is shown in Equation 1.

$$\gamma(\alpha) = \frac{N_c}{N} \qquad (1)$$

Where N_c : The number of documents in class α

N : Total number of documents

In Multinomial model, assumes attribute values are

independent of each other given for the particular class $\gamma(\beta \mid \alpha) = \gamma(\omega_{1}...\omega_{nd} \mid \alpha)$. In the multinomial model, a document is an ordered sequence of word events, drawn from the same vocabulary V. Assume that the lengths of documents are independent of class. Thus, each document β_i is drawn from a multinomial distribution of words with as many independent trials as the length of β_i. This yields the familiar bag-of-words representation for documents. The BOW model [5] is commonly used in methods of document classification, where the (frequency of) occurrence of each word is used as a feature for training a classifier. From given documents, a predefined list of words appearing in the documents (the dictionary) and computes the vectors of frequencies of the words as appear in the documents. The angle between the two vectors is a widely used measure of closeness between documents. Let W be the dictionary - the set of all terms (words) that occur at least once in a collection of documents D. The BOW representation of document d_n is a vector of weights $(\omega_{1n}...\omega_{|W|n})$. In the simplest case, the weights $\omega_{in} \in 1,-1$ and denote the presence or absence of a particular term in a document. More commonly, ω_{in} represent the frequency of the ith term in the nth document, resulting in the term frequency representation. The transformation of a document set D into the BOW representation enables the transformed set to be viewed as a matrix, where rows represent document vectors, and columns are terms. A unigram feature marks the presence or absence of a single word within a text. Estimate the conditional probability $\gamma(\omega \mid \alpha)$ as the relative frequency of term ω in documents belonging to class α including multiple occurrences of a term in a document.

$$\gamma(\omega \mid \alpha) = \frac{\text{count}(\omega, \alpha) + 1}{\text{count}(\alpha) + |V|} \qquad (2)$$

Where $\text{count}(\omega, \alpha)$: Number of occurrences of ω in training documents from class α

$\text{count}(\alpha)$: Number of words in that class

$|V|$: Number of terms in the vocabulary

The problem with the MLE estimate is that it is zero for a term-class combination that did not occur in the training data. The training data are never large enough to represent the frequency of rare events adequately. To eliminate zero probability problem [18], use add-one or Laplace smoothing, this simply adds one to each count. Add-one smoothing can be interpreted as a uniform prior (each term occurs once for each class) that is then updated as evidence from the training data comes in. Then, the probability of a document given its class is simply the multinomial distribution presents in Equation 2. Finally classify new document using posteriori probability. Let α_{NB} is the posterior probability, α_j is one of the class from class α and β_i is ith document.

$$\alpha_{NB} = \arg \max_{\alpha_j \in \alpha} \prod_{i} \gamma(\beta_i \mid \alpha_j)$$

Some researches show that although the assumption independence between words in a document is not fully met, it is obvious that the conditional independence assumption is rarely true in most real-world applications. A straightforward approach to overcome the limitation of Naive Bayes is to combine with Neural Network to represent explicitly the dependencies among attributes. Table 1 take as the example of movie review. Using Naive Bayes classifier classifies movie review in positive or negative category. Calculate priori probability of pos and neg by using Equation 1

$$\gamma \text{ (pos)} = 3/4$$

$$\gamma \text{ (neg)} = 1/4$$

Calculate maximum likelihood smoothing Naive Bayes estimate by using Equation 2

γ (like | pos) = (3 + 1) / (25 + 31) = 4/56 = 1/14

γ (boring | pos) = (0 + 1) / (25 + 31) = 1/56

γ (good | pos) = (1 + 1) / (25 + 31) = 2/56 = 1/28

Table 1: Example

Set	Document	Words	Class
Training set	1	I like movie. It's story nice.	pos
	2	Hero's acting is good, I like it but heroin role is bad. Overall movie is fantastic.	pos
	3	I like music, which is so rocking.	pos
	4	Movie story is good but ending is just plain boring and sadly.	neg
Test set	5	I like director's direction. The location place in movie is so boring. But still movie is good.	?

γ (like | neg) = (0 + 1) / (12 + 31) = 1/43

γ (boring | neg) = (1 + 1) / (12 + 31) = 2/43

γ (good | neg) = (1 + 1) / (12 + 31) = 2/43

Calculate posteriori probability

γ (pos | d5) = 3/4 * 1/14 * 1/56 * 1/28 = 3.4165e- 5

γ (neg | d5) = 1=4 * 1/43 * 2/43 * 2/43 = 1.2577e- 5

γ (pos | d5) > γ (neg | d5)

γ (pos | d5) is maximum means probability of positive words in document 5 is maximum so document 5 is positive.

3.3 Neural Network Classifier

Neural Network has emerged as an important tool for classification. The recent vast research activities in neural classification have established that Neural Networks are a promising alternative to various conventional

classification methods. The Neural Network with appropriate network structure handles the correlation or dependence between input variables. The advantage of Neural Networks [19] lies in the some theoretical aspects. First, Neural Networks are data driven self-adaptive methods in that they can adjust themselves to the data without any explicit specification of functional or distributional form for the underlying model. Second, Neural Networks are a nonlinear model, which makes them flexible in modeling real world complex relationships. Finally, Neural Networks are able to estimate the posterior probabilities, which provide the basis for establishing classification rule and performing statistical analysis. Neural Networks [20] have been successfully applied to a variety of real world classification tasks in industry, business and science.

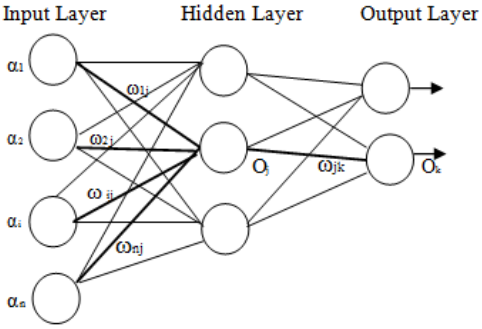

Figure 2: Structure of Neural Network

Figure 2, shows the structure of Neural Network [21]. Neural Network constructs using input layer, hidden layer and output layer. Each layer is made up of units. The inputs to the network correspond to the attributes measured for each training tuple. The inputs are fed simultaneously into the units making up the input layer. These inputs pass through the input layer and are then weighted and fed simultaneously to a second layer of neuron like units, known as a hidden layer. The outputs of the hidden layer units are input to another hidden layer and so on. Each output unit takes, as input, a weighted sum of the outputs from units in the previous layer. Neural network is used for both classification, to predict the class label of a given document or prediction, to predict a continuous-valued output. For classification, one output unit may be used to represent two classes for example the value 1 represents one class and the value 0 represents the other. If there are more than two classes, then one output unit per class is used. Let α and β be an input and output of input, hidden and output layer respectively.

$$\beta_i = \alpha_i \quad (3)$$

Where α_i : input of an input unit i in input layer

β_i : Output of an Input unit i in input layer

$$\rho\alpha_j = \sum_i \omega_{ij}\ \beta_j \quad (4)$$

Where ω_{ij} : Weight from unit i to j

β_j : Output of an Input or Hidden unit j in Input layer or Hidden layer respectively $\rho\alpha_j$: Input of Hidden unit or Output unit j in Hidden layer or Output layer respectively

$$\partial\beta_j = \frac{1}{1+e^{-\lambda\rho\alpha j}} \quad (5)$$

Where $\rho\alpha_j$: Input of hidden unit or output unit j in Hidden layer or Output layer respectively $\partial\beta_j$: Output of hidden unit or output unit j in Hidden layer or Output layer respectively For example, classification is done using Neural Network classifier which takes example from Table 1. First assign input values to input layer and the weights in the network are initialized to small random numbers (e.g., ranging from -0.1 to 1.0 or -0.5 to 0.5) and assume λ =1. Figure 3 shows the initial values of input and weight of Neural Network.

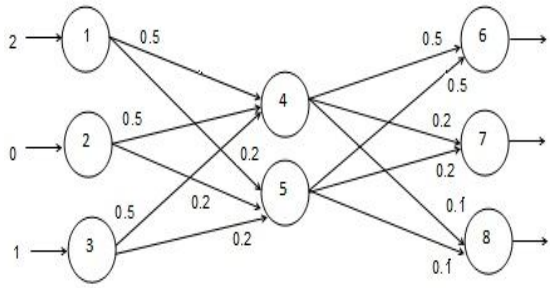

Figure 3: The initial values of input and weight of Neural Network.

To calculate Output of Input layer by using Equation

$\beta_1 = \alpha_1 = 2$

$\beta_2 = \alpha_2 = 0$

$\beta_3 = \alpha_3 = 1$

To calculate Input of Hidden Layer by using Equation 4

$\rho\alpha_4 = \omega_{14}\ \vartheta\beta_1 + \omega_{24}\ \vartheta\beta_2 + \omega_{34}\ \vartheta\beta_3$

$= 0.1 * 2 + 0.1* 0 + 0.1 * 1 = 0.3$

$\rho\alpha_5 = \omega_{15}\ \vartheta\beta_1 + \omega_{25}\ \vartheta\beta_2 + \omega_{35}\ \vartheta\beta_3$

$= 0.2 * 2 + 0.2 * 0 + 0.2 * 1 = 0.6$

To calculate Output of Hidden Layer by using Equation

$$\rho\beta_4 = \frac{1}{1+e^{-0.3}} = 0.57444$$

$$\rho\beta_5 = \frac{1}{1+e^{-0.6}} = 0.64565$$

To calculate Input of Output layer by using Equation 4

$\partial\alpha_6 = \omega_{46}\ \rho\beta_4 + \omega_{56}\ \rho\beta_5$

$= 0.5 * 0.57444 + 0.5 * 0.64565$

$= 0.610045$

$\partial\alpha_7 = \omega_{47}\ \rho\beta_4 + \omega_{57}\ \rho\beta_5$

$= 0.2 * 0:57444 + 0.2 * 0.64565$

$= 0.24401$

$\partial\alpha_8 = \omega_{48}\ \rho\beta_4 + \omega_{58}\ \rho\beta_5$

$= 0.1 * 0.57444 + 0.1 * 0.64565$

$= 0.639001$

To calculate Output of Output Layer by using Equation 5

$$\partial\beta_6 = \frac{1}{1+e^{-0.610045}} = 0.647951$$

$$\partial\beta_7 = \frac{1}{1+e^{-0.24401}} = 0.560701$$

$$\partial_{\beta 8} = \frac{1}{1+e^{-0.6390014}} = 0.6545276$$

Output = [0.647951 0.560701 0.6545276]
Output = [Positive Neutral Negative]
Predicted output is negative. The output matrix selects maximum value as output. So in given example test file is negative.

3.4 Naive Bayes Neural Classifier

The Naive Bayes Neural classifier is a classifier whose output is based on Naive Bayes and Neural Network classifier output. The output of Naive Bayes and Neural Network classifier is differentiated by using confusion matrix.

Table 2: Confusion Matrix Structure

		Expected Output
Predicted Output	Correct output	Incorrect output
	Incorrect output	Correct output

A confusion matrix is a visualization method typically used in supervised learning. Table 2 shows the structure of confusion matrix. Each column of the matrix represents the instances in an expected class, while each row represents the instances in a predicted class. One benefit of a confusion matrix is that it is easy to see if the system is confusing two classes. When a data set is unbalanced (when the number of samples in different classes vary greatly) the error rate of a classifier is not representative of the true performance of the classifier. The diagonal of confusion matrix is indicated the correct classifications. The remaining values of confusion matrix is indicated the incorrect classifications. The size parameter is used for differentiating Naive Bayes classifier from Neural Network. The size of confusion matrix of Naive Bayes and Neural Network represented by $\tau(\text{sna})$ and $\sigma(\text{snn})$ respectively. Correct output of classification is assign to $\psi(\text{out})$. Let α and β be the output of Naive Bayes and Neural Network classifier respectively. Either of following statement is true.

(1) If $\tau(\text{sna}) = 1$ then $\psi(\text{out}) = \alpha$

(2) But if $\tau(\text{sna}) \neq 1$ and $\sigma(\text{snn}) = 1$ then $\psi(\text{out}) = \beta$

(3) But if $\tau(\text{sna}) \neq 1$ and $\sigma(\text{snn}) \neq 1$ then $\psi(\text{out}) = \alpha$

Table 3 shows values which Naive Bayes classifier gives predicted output is 1 and expected output is 1 for example shown in Table 1.

Table 3: Confusion Matrix of Naive Bayes Classifier

		Expected Output	
		-1	1
Predicted Output	-1	0	0
	1	0	1

Confusion matrix returns correct output as 1 means expected and predicted output is same, where Neural Network gives predicted output as a -1 and expected output as a 1 of given example, shown in Table 4.

Table 4: Confusion Matrix of Neural Network Classifier

		Expected Output	
		-1	1
Predicted Output	-1	0	0
	1	1	0

From Table 3 shows Naive Bayes gives expected and predicted output is same. Confusion matrix of Naive Bayes returns correct output and size is $\tau(\text{sna}) = 1$ and from Table 4 gives incorrect output and that time size is $\sigma(\text{snn}) = 2$. So Naive Bayes Neural *classifier* gives output of Naive Bayes. Naive Bayes Neural classifier is classifying such combining correct output of training dataset

4. RESULT

The movie review dataset is used for this work that is provided by [4]. Experimental setup contains simulation environment, parameters and performance metrics. Generally performance metrics are used for calculate one of the metrics like size, execution time, performance accuracy of system.

4.1 Simulation Environment and Parameters

The proposed system performance is evaluated using MATLAB R2012a environment. The keyword dictionary and movie review dataset is input to proposed system. The WordStat Dictionary is downloaded from Bill McDonald's website [15]. It is commonly used dictionary in sentiment analysis. It contains positive and negative words. This dictionary has 2683 total keywords which includes 354 positive words and 2396 negative words. For experimental work, select a ready-to-use and clean dataset of movie reviews domain [14]. The movie review dataset contains 2000 text files in which 1000 labeled as positive reviews and the rest 1000 labeled as negative reviews.

4.2 Performance Metrics

Performance metrics are used for the analysis of classifier accuracy. The proposed system is evaluated performance using accuracy parameter. Accuracy is calculated by Equation 6.

$$\text{Accuracy} = 100 - \frac{\text{Number of InCorrect Sample} * 100}{\text{Total Number of sample}} \quad (6)$$

4.3 Experimental Result

In experiment, Naive Bayes classifier and Neural Network classifier is used for classifying the polarity of the documents in the dataset. In training phase, train the classifiers with features for movie review dataset. A unigram feature is used to marks the presence or absence of a single word within a text. In experiment, a bag of words was represented by a vector containing values indicating the number of times each feature occurred in the document. After conducting experiment, the proposed system is shown polarity classification result on given

movie review. In which test one of the review from dataset and checks that review is positive or negative polarity. Table 5 shows result after experimented system which found 1247 correct samples from 2000 review using Naive Bayes classifier. Correct samples 999 are found by Neural Network classifier. From combing result of Naive Bayes and Neural Network classifier, Naive Bayes Neural classifier found 1613 correct samples.

Table 5: Result of Experiment

Total No. of Reviews	Classifier	Correct Sample	Incorrect Sample
2000	Naive Bayes classifier	1247	753
	Neural Network classifier	999	1001
	Naive Bayes Neural classifier	1613	387

5. DISCUSSION

The classification is done by using data mining techniques that are Naive Bayes classifier and Neural Network Classifier. In training phase, the Naive Bayes classifier is trained. Movie review test file is tested using trained classifier. The polarity classification is shown as the result of proposed system. That may be either positive or negative review. In Table 6, shows the accuracy results of classifier. Accuracy is calculated by using Equation 6. After experiment, the result of sentiment analysis using Naive Bayes classifier is obtained 62.35% accuracy on training data because of its strong independence assumptions among features independent to other features given the class.

Table 6: Accuracy of Classifier

Dataset	**Feature**	**Classifier**	**Performance Accuracy of Classifier**
Movie Review Dataset	Unigram (Bag of words)	Naive Bayes Classifier	62.35%
		Neural Network Classifier	49.95%
		Naive Bayes Neural Classifier	80.65%

Table 6 shows accuracy of sentiment analysis using Naive Bayes classifier with unigram feature. The Neural Network with appropriate network structure handles the correlation between input variables. Using Neural Network classifier, the accuracy of sentiment analysis is obtained 49.95% on training dataset. Neural Network provides better result in complex domain. The sentiment analysis using Naive Bayes Neural classifier is giving the correct output. The correct output is obtained by combining the predicted output of Naive Bayes result and Neural Network classifier result along with actual output using confusion matrix. From assumptions of dependency and independency among features, result of Naive Bayes and Neural Network classifier is improved by using Naive Bayes Neural classifier. Table 6 shows accuracy of sentiment analysis using Naive Bayes Neural classifier, which is 80.65% for training movie review dataset.

6. CONCLUSION

Sentiment analysis play vital role to make decision like movie review domain. Due to its tremendous value for practical applications, there has been an explosive growth of both research in academia and applications in the industry. In this study, an experiment is conducted on Movie Review dataset. The Naive Bayes classifier and Neural Network classifier is used to train dataset. Movie review is classified by using Naive Bayes, Neural Network and Naive Bayes Neural classifier. Accuracy of sentiment analysis is increased by proposed system from dependence and independence assumptions among features. In future, apply this work on clustering domain for movie review dataset for opinion mining applications where the cluster based features are used to address the problem of scarcity of opinion annotated data in a language.

References

[1] N. M. Shelke, S. Deshpande, V. Thakre, "Survey of techniques for opinion mining," International Journal of Computer Applications, vol.57, no.13, pp. 30-35, November 2012.

[2] B. Seerat and F. Azam, "Opinion mining: Issues and challenges (a survey)," International Journal of Computer Applications, vol. 49, no. 9, pp. 42-51, July 2012.

[3] P. Routray, C. K. Swain, S. P. Mishra, "A survey on sentiment analysis," International Journal of Computer Applications, vol. 76, no. 10, pp. 1-8, August 2013.

[4] B. Pang, L. Lee, S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 79-86. July 2002.

[5] M.Radovanovic, M. Ivanovic, "Text mining: Approaches and applications," vol. 38, no. 3, pp. 227-234,[Online]. Available:http://www.emis.de/journals/NSJOM/Papers/383/NSJOM 3[Accessed: December 2013].

[6] S. Deng, T. Mitsubuchi, K. Shioda, T. Shimada, A. Sakurai, "Combining technical analysis with sentiment analysis for stock price prediction," in Ninth IEEE International Conference on Dependable, Autonomic and Secure Computing, 2011, pp. 800-807.

[7] Y. Qiu, G. Yang, and Z. Tan, "Chinese text classification based on extended nave bayes model with weighed positive features," in First International Conference on Pervasive Computing, Signal Processing and Applications, pp. 243-246, 2010.

[8] W. Duan, Q. Cao, Y. Yu, "Mining online user-generated content: Using sentiment analysis technique to study hotel service quality," in 46th Hawaii International Conference on System Sciences, pp. 3119-3128, 2013.

[9] T. Xu, M. Xu, H. Ding, "Bbs topics hotness forecast based on back-propagation neural network," in International Conference on Web Information Systems and Mining, pp. 57-61, 2010.

[10] P. Gamallo, M. Garcia, Santiago, "Tass: A naive-bayes strategy for sentiment analysis on spanish tweets," in International Conference on Social Informatics, pp. 215-221, 2012.

[11] M. A. Fattah, "Gaussian mixture model and feed forward neural network based models for sentiment classification," In International Conference on Computer Science and Information Technology, pp. 45-49, 2012.

[12] M. Kaya, G. Fydan, I. Toroslu, "Sentiment analysis of turkish political news," in IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, pp. 174-180, 2012.

[13] W. Simm, M. A. Ferrario, S. Piao, J. Whittle, P. Rayson, "Classification of short text comments by sentiment and actionability for voiceyourview, " in IEEE International Conference on Social Computing /IEEE International Conference on Privacy, Security, Risk and Trust, pp. 552- 557, 2010.

[14] "Movie review dataset," [Online]. Available http://www.cs.cornell.edu/people/pabo/movie-review-data/, [Accessed: October 2013].

[15] B. McDonald, "Keyword dictionary," [Online]. Available: http://www3.nd.edu/mcdonald/WordLists.html, [Accessed: October 2013].

[16] C. Tseng, N. Patel, H. Paranjape, T. Y. Lin, S. Teoh, "Classifying twitter data with naive bayes classifier," in IEEE International Conference on Granular Computing, 2012.

[17] K. M. Leung, "Naive Bayesian classifier," [Online]. Available:http://www.sharepdf.com/81fb247fa7c54680a94dc0f3a253fd85/naiveBayesianClassifier.pdf, [Accessed: September 2013].

[18] H. Shimodaira, "Text classification using naive bayes," pp. 1-5, [Online].Available: http://www.inf.ed.ac.uk/teaching/courses/inf2b/learnnotes/inf2blearn-note07-2up.pdf, [Accessed: December 2013].

[19] G. P. Zhang, "Neural networks for classification: A survey," in IEEE Transactions on Systems, MAN and Cybernetics Part C: Applications and Reviews, vol. 30, no. 4, pp. 451-462, November 2000.

[20] W. Claster, D. Q. Hung, S. Shanmuganathan, "Unsupervised artificial neural nets for modelling movie sentiment," in Second International Conference on Computational Intelligence, Communication Systems and Networks, pp. 349-354, 2010.

[21] J. Han, M. Kamber, Data Mining Concepts and Techniques, 2nd ed. Morgan Kaufmann, 2006.