



Sentiment Analysis and Prediction Using Neural Networks

Sneh Paliwal^(✉), Sunil Kumar Khatri, and Mayank Sharma

Amity Institute of Information Technology, Amity University,
Noida, Uttar Pradesh, India

sneh.paliwal@student.amity.edu, sunilkhatri@gmail.com
{skkhatri, msharma22}@amity.edu

Abstract. Sentiment Analysis or opinion mining have taken many leaps and turns from its starting in early 2000s till now. Advancement in technology, mobile & internet services and ease of access to these services have resulted in more and more engagement of people on the social media platforms for expressing their views and collaborate with people who share similar thoughts. This has led to generation of a large amount of data on the internet and subsequently the need of analysing this data. The sentiment analysis helps different organisations to know how people look to their products and services and what changes are required to improve them. The paper performs sentiment analysis i.e. classification of tweets into positive, negative and neutral on views of a particular product using an inbuilt python library called TextBlob for three platforms i.e. twitter, Facebook and news websites and further it talks about how Artificial Neural Networks (ANN) offer a platform to perform sentiment analysis in a much easier and less time-consuming manner. In this paper Feed-Forward Back propagation neural networks are used to split the data into train and test data and a min-max approach was applied to the data to scale the data and analyse the prediction accuracy of a sentiment using ANN. Precision, recall and accuracy have been calculated to provide a quantitative approach to the results and measure the performance of ANN. We found that such type of neural network is very efficient in predicting the result with a high accuracy.

Keywords: Sentiment analysis · Artificial Neural Networks (ANN)
Machine learning · Feed-forward networks · Activation function

1 Introduction

Sentiment Analysis or more commonly called opinion mining has been helping different organisations and industries to analyse and predict trends for nearly two decades. With increase in access to internet and its services more and more people are joining and participating in social media activities. This has led to generation of a huge amount of data on the social media platforms like twitter, Facebook, YouTube etc. Today most of the organisations analyse user sentiments through this social media content. It helps them to improve their business decisions and hence maximise profits. Many scientists have been working on different methods to classify and analyse the data generated through these platforms. Some of the methods include Naïve Bayes theorem, Support

vector Machine (SVM), Maximum entropy etc. These algorithms have been proved to be very efficient with data which is smaller in size. But the increase in the amount of data in terms of velocity, veracity and volume have generated a need for much efficient algorithms which could analyse a large amount of data and provide a greater precision. In this paper we are performing sentiment analysis for 'iPhoneX' using twitter API, Facebook API and data collected from news websites using python as a platform and around 36,500 tweets, comments and votes from each platform have been collected and cleaned. The polarity and sentiment of the tweets was recorded using the inbuilt TextBlob library. Further ANN are applied to the collected data using R. The data collected is partitioned into train and test dataset in the ratio 80:20 respectively. The train data is used to train the neural network and test dataset is used to determine the accuracy of the results predicted by ANN. This paper is categorized as follows: Sect. 2 gives an overview of the literature survey. Section 3 states the proposed approach. Section 4 gives the methodology adopted. Section 5 tabulates the results. Conclusion is stated in Sect. 6 of the paper.

2 Related Work

Much research work is done on classifying data and performing sentiment analysis using the traditional methods of sentiment analysis like Naïve Bayes, Lexicon based approach, Support Vector Machine (SVM), Maximum entropy etc. The authors have used twitter data to find and analyze the sentiment of the tweets. After collecting the tweets, score of the tweets were calculated and finally a performance analysis was done. The authors have used Random Sampling technique to select a sample dataset from the whole dataset. On the same grounds, Sampling Analysis and comparative analysis was performed for sample dataset and between sample and whole dataset respectively [1]. In this paper a module is trained with the help of Hadoop and MapReduce, Classification was based on Naïve Bayes, Time Variant Analytics and the Continuous learning System. Real time analysis done in this paper was a major highlight of this paper [2]. In this paper, the authors have tried to implement dictionary based methodology of sentiment analysis and in order to estimate the sentiment of the public an algorithm was developed that is employed to large amount of data. An acronym dictionary was used by the authors to identify the acronyms and moreover emoticons were also detected from the tweet. They have done both document level and aspect level analysis [3]. In this paper the authors have tried to ease information extraction from financial documents available online. For the above job the authors have classified the text on the basis of genre and sentiment. To classify text on the basis of genre, Support vector Machine has been used and to classify the text on the basis of sentiment, an attribute extraction algorithm has been designed on the basis of Apriori Algorithm to extract attributes from the text and design a lexicon. The results were satisfactory when authors have used sentiment intensity calculation [4]. In this paper, the author has implemented two algorithms i.e. Naïve Bayes (NB) and Maximum Entropy. Through the observations it was very clear that maximum Entropy classifier is better than any other classifier for predicting the sentiment with 74% accuracy [5]. Through this paper the authors have performed sentiment analysis on the comments of

the users on YouTube videos by using Natural Language Processing (NLG) and SentiStrength to improve the relevancy and quality of YouTube videos. Around one million user comments have been used to perform the analysis and the authors have shown the efficiency of up to 75.4% [6]. The experiment performed in this paper indicated that SVM shows better results as compared to Naive Bayes and Maximum Entropy classifiers. The observation showed accuracy of SVM is 91%, the accuracy of Naive base is 83% and the accuracy of Maximum Entropy is 80% [8]. In this paper the author is trying to compare various classification algorithms such as Random Forest, Gradient Boosting, Decision Tree, Adaptive Boost, Logistic Regression and Gaussian Naïve Bayes to recognize sarcasm in tweets from the Twitter Streaming API [9]. In this paper the authors have performed sentiment analysis on the reviews of the customers given for different mobile phones. The authors have included the sentiments like anger, disgust, fear, joy, trust etc. to classify the text as positive or negative. The authors have performed a sentence level classification of the reviews by using an inbuilt package named ‘Syuzhet’. The results after being cross validated using Support Vector Machine (SVM) give an accuracy of 84.87% [10]. In this paper the authors have performed sentiment analysis on Delhi Corporation Election results by using different supervised machine learning classifiers. Through this paper authors are trying to figure out the most effective and accurate algorithm for prediction. The results show that Multinomial Naïve Bayes classifier is the most effective predictor for sentiments with an accuracy of 78% [11]. Authors have used Rapid Miner to implement machine learning algorithms and perform opinion mining. The Support Vector Machine algorithm classified the text into positive, negative and neutral depending on their polarity [12]. In this paper authors have made a comparative analysis of various algorithms and suggested that among traditional algorithms used so far SVM has given better results but also points towards the limitations of using SVM and suggests that the use of artificial neural networks (ANN) in sentiment classification and analysis can overcome these problems. The authors have concluded that machine learning approaches and ANN implementations would result in better classification and analysis. A methodology has also been proposed which suggests how ANN can be applied to large datasets and better results can be obtained [14]. In the proposed work, ANN has been applied to different datasets obtained from social media platforms like twitter, Facebook and reviews of a product on news websites, to check the performance of ANN and supervised learning approach (which is used to train the dataset obtained) by calculating precision, recall and accuracy.

3 Proposed Approach

A. Artificial Neural Networks and Their Need

An artificial neural network (ANN) is a computational model in light of the structure and elements of natural or biological neural systems. Structure of an ANN is based in the data that traverses through the network as based on the fact that a neural network, whether it is artificial or biological, learns according to the data that is fed into it and the output it gives.

ANNs are considered nonlinear statistical data modelling tools in which the intricate relationship between inputs and outputs are used to recognise or identify some patterns. Instead of using the whole dataset, ANN's use a sample of the data to find and reach the solution so this saves both time and money in finding and arriving at the result. ANNs are regarded as fairly simple mathematical models to upgrade the present data analysis technologies. Just like there are neurons in the brain for passing information, there are nodes in a neural network for doing a similar task. Nodes are nothing but mathematical functions. ANNs have three interconnected layers just like a neuron in our body. The first layer is similar to dendrites where the input value is received. The second layer is analogous to soma where the summation of the information is done and the output is given at the last i.e. the output layer which is like the axon of a neuron (Fig. 1).

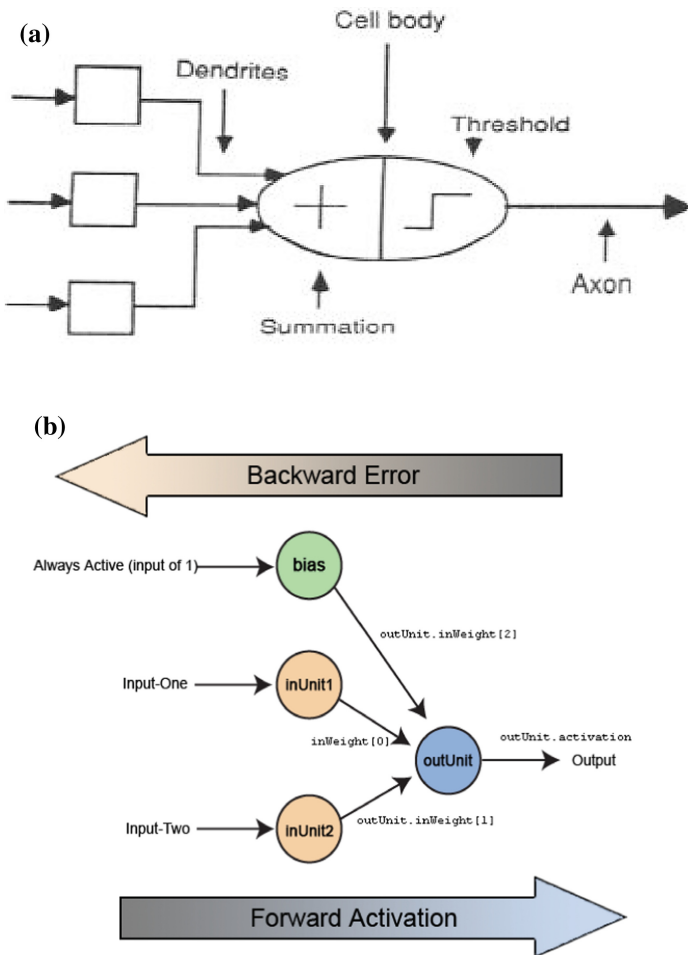


Fig. 1. (a) Artificial neural networks and similarity with neurons [15]. (b) Back propagation model [16]

Components of Artificial Neural Networks:

Neurons: Each neural network comprises of several neurons which constitutes an input neuron, inner neurons and an output neuron. The value in each neuron is stored depending on the predecessor neuron (except the input neuron) and is forwarded to the successor neuron after the applying the activation function.

Connections and weights: Two neurons form a connection between them and each connection is assigned a weight.

Propagation function: This function computes the input to the next neuron by processing the weight from the connection.

Learning Rule: This learning rule implies an algorithm that modifies the weights and threshold variables for a neural network.

B. Types of Neural Networks

A neural network is of two types:

1. Convolutional neural network.
2. Artificial neural network.

A convolutional neural network is primarily used for tasks related to image processing like classification of images into various groups, presence of a tumour etc. Convolutional networks perform optical character recognition (OCR) to digitize text and make natural-language processing possible on analogue and hand-written documents, where the images are symbols to be transcribed. CNNs can also be applied to sound when it is represented visually as a spectrogram.

An artificial neural network is used for all the other machine learning tasks other than related to image processing, which generally involve number crunching (like stock prediction, to predict whether a certain team would win a match or not based on its performance etc.)

Feed-Forward Neural Networks: The feed forward neural network or a multilayer perceptron are neural networks where the information or values that are fed into them flows in one direction i.e. there are no outputs which are fed back to the network. There are no cycles or loops in the network. If a loop is present then such networks are called recurrent neural networks. This type of neural network is used to test the data after training.

Back Propagation Algorithm: In this algorithm a small value is back propagated to the hidden layer so that the system can train itself. This is used in training the dataset. The Backpropagation algorithm looks for the minimum value of the error function in weight space using a technique called the delta rule or gradient descent. The weights that minimize the error function is then considered to be a solution to the learning problem. The “backwards” part of the name comes from the fact that calculation of the gradient proceeds backwards through the network, with the gradient of the final layer of weights

being calculated first and the gradient of the first layer of weights being calculated last. Partial computations of the gradient from one layer are reused in the computation of the gradient for the previous layer. This backwards flow of the error information allows for efficient computation of the gradient at each layer versus the naive approach of calculating the gradient of each layer separately.

C. Types of Learning

Supervised Learning: This is a machine learning task where a function is learnt by the machine by mapping input and output pairs. It requires a training dataset which provide the machine with required input and output pairs. In this type of learning the system is provided the output in order to train it and then it performs the prediction operation. It is similar to training a child. In the same way, the system first learns about the input-output pairs and then predicts the output whenever it encounters the same input.

Unsupervised Learning: This type of learning is also a machine learning algorithm which is used to draw inference from a dataset. It takes data without responses as input so there is no measure of accuracy. His learning technique do not provide the system with output and the system marks them on the basis of some characteristics like shape, colour, size etc. and the output is predicted on this basis.

4 Methodology

Using the twitter API and python as a tool, 36,500 tweets were collected and filtered by removing unwanted words, special characters and spaces. A dataset was created which include date, number of positive tweets, number of negative tweets, number of neutral tweets, polarity. The polarity of the tweets was determined by an inbuilt library in python i.e. TextBlob.

TextBlob is a Python library for processing textual data. An API is provided to perform sentiment analysis. When a text is fed as input, it uses Naïve Bayes classifier to classify the text. This analyser is apparently based on Stanford Natural Language Tool Kit (NLTK). The training data for Stanford NLTK are the movie reviews.

The same procedure is applied for collecting data from Facebook and news websites. Using R programming, the data was partitioned as train and test dataset in the ratio of 80:20 respectively. The train dataset was given to a feed forward neural network which uses supervised learning approach to train the neural network. A neural network graph was then created with three inputs to the input layer which include number of positive tweets, number of negative tweets and number of neutral tweets. The second layer is the hidden layer and the third layer is the output layer which determines the overall polarity of the data that is fed into the neural network. A confusion matrix was then created to calculate the accuracy of the results obtained (Fig. 2).

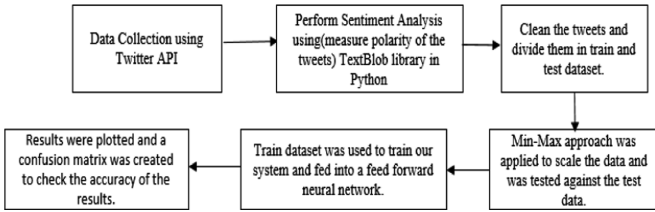


Fig. 2. Flowchart of the approach

Also, to validate the results precision and recall values are computed for all the three datasets used in the study.

A. Implementation

Using the twitter application interface, Facebook graph API’s and web scraping using python as the programming language, last one year tweets, comments and votes related to ‘iphoneX’ were collected limiting the number of tweets per day to be 100 which resulted in a total of 36,500 tweets, 36,500 comments and 36,500 votes for performing data analytics. The data collected from twitter includes the date, text of the tweet and its sentiment by using twitter API. Similarly, data from Facebook is collected using Facebook’s graph API and data from news websites is collected using web scraping. After that the data was cleaned by removing unwanted symbols and spaces. The collected data was analyzed using TextBlob library and sentiment of the data was noted. Now the number of positive, negative and neutral tweets, comments and votes were recorded per day with the overall polarity of the day. The polarity of the tweets, comments and votes was calculated by subtracting number of positive and the negative data. The data collected contains date of the tweet, No. of positive tweets, No. of negative tweets, No. of neutral tweets, polarity of the data and scaled polarity whose value lies between 0 and 1.

To apply the neural networks, R programming language has been used to predict and analyze the results. The dataset was divided into train and test dataset in the ratio of 80:20 respectively. The train dataset is now scaled using min-max approach to scale the values between 0 and 1. The Fig. 3 shows a sample of scaled data.

	No_of_positive_tweets	No_of_negative_tweets	No_of_neutral_tweets
1	0.1126760563	0.0000000000	0.3510638298
6	0.5211267606	0.1666666667	0.7553191489
13	0.4647887324	0.6666666667	0.1382978723
27	0.4084507042	0.8333333333	0.8510638298
31	0.2957746479	1.0000000000	0.5212765957
36	0.4084507042	0.0833333333	0.3829787234

Fig. 3. Scaled data

From the train dataset number of positive, negative and neutral tweets were fed into the input layer. Similar work is done for data from Facebook and news websites. The neural network applied here is a feed forward neural network and uses back propagation to train itself. The train dataset is required to train our neural networks and the predicted results will be tested against the test dataset. After training the neural network a neural net graph was plotted which is shown in Fig. 4.

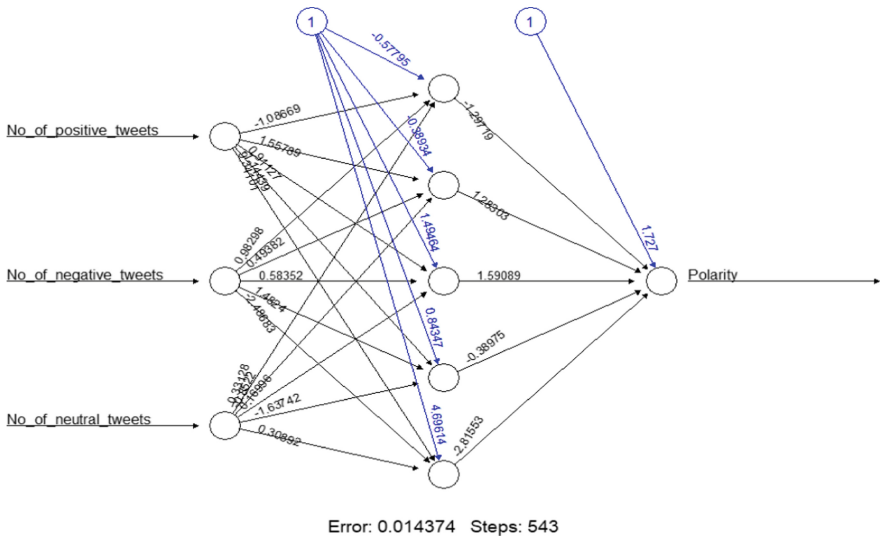


Fig. 4. Artificial neural network graph for twitter data

The Fig. 4 has an input layer which provides three inputs to the neural network i.e. the number of positive, negative and neutral tweets per day to the neural network and gives polarity of the tweet as a result in the output layer. Similar neural nets were plotted for Facebook and new websites data. Similar figures shows neural nets for Facebook and news websites data. Input for Facebook data were the number of positive, negative and neutral comments and for news website data the input consist of number of up votes, down votes and neutral votes and the output of all the neural nets was polarity of the input data (Fig. 5).

The test dataset was now fed to the trained neural network and a matrix was taken as output which showed actual and predicted values of the polarity (Fig. 6).

Program Code: The code used for plotting the neural network graph is given below:

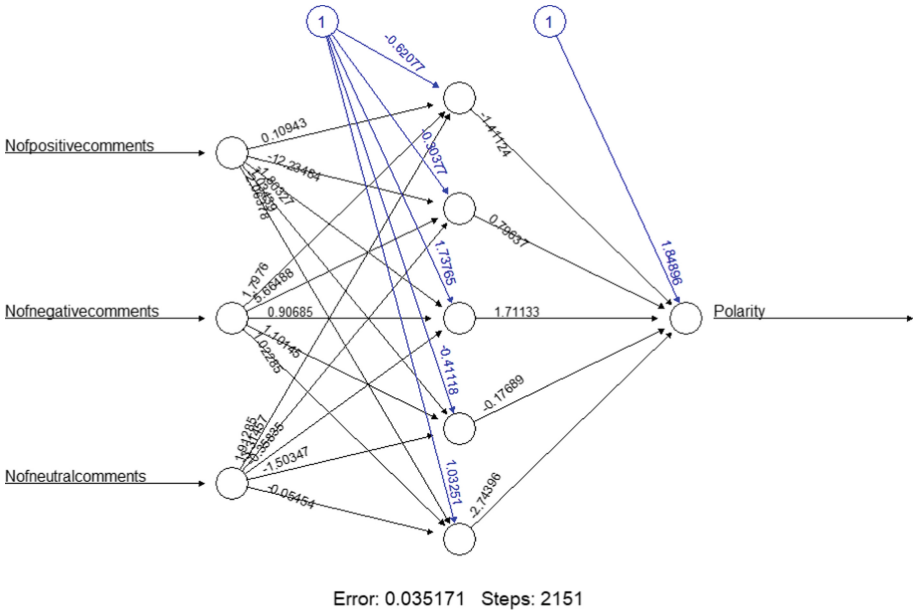


Fig. 5. Artificial neural network graph for Facebook data

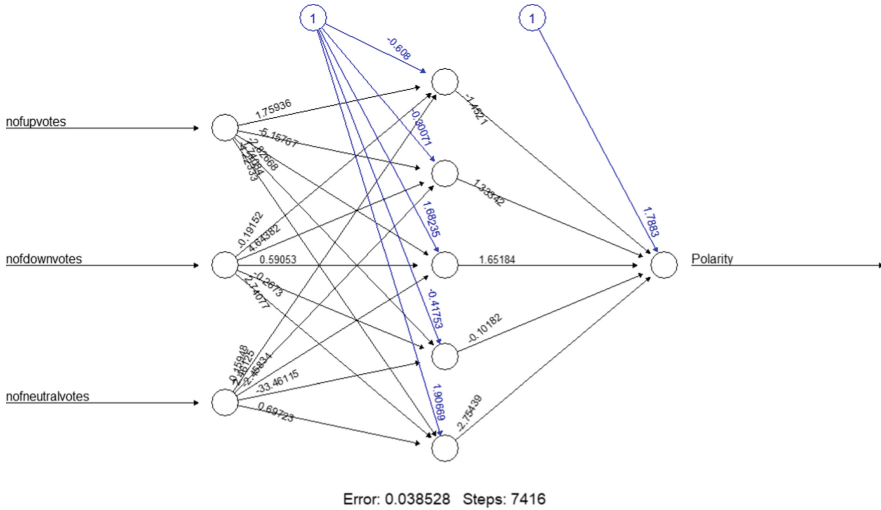


Fig. 6. Artificial neural network graph for news website data

```

getwd()
setwd('C:/Users/Sneh')
getwd()
data = read.csv("perDate_Sentiment.csv", header=T)
samplesize = 0.80 * nrow(data)
#set.seed(80)
index = sample( seq_len ( nrow ( data ) ), size = samplesize )
datatrain = data[ index, ]
datatest = data[ -index, ]
data <- data[, sapply(data, is.numeric)]
maxs <- as.numeric(apply(data, 2, max) )
mins <- as.numeric(apply(data, 2, min))
scaled <- as.data.frame(scale(data, center = mins, scale = maxs - mins))
library(neuralnet)
trainNN = scaled[index , ]
testNN = scaled[-index , ]
set.seed(2)
NN = neuralnet(Level ~ No_of_positive_tweets+ No_of_negative_tweets+
No_of_neutral_tweets, trainNN, hidden = 5 , linear.output = T )
plot(NN)

```

Now the actual and predicted values are rounded to one place of decimal and a confusion matrix was plotted. With the help of confusion matrix we calculate the precision, recall and accuracy of the result. Figures 7(a), (b) and (c) shows the confusion matrices.

The following formulae depicted in Eqs. 1, 2 and 3 are used to calculate Precision, Recall and Accuracy. Precision measures how often a sentiment is rated correctly. Recall measures how many tweets, comments or votes with sentiment are rated as sentimental. This also determines how well ANN's can identify neutrality. The third parameter i.e. accuracy checks how many sentiments that were rated as positive or negative were rated correctly.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (1)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (2)$$

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{TN} + \text{FN}) \quad (3)$$

where

TP = True Positive
 FP = False Positive
 TN = True Negative
 FN = False Negative

A similar confusion matrix was constructed for Facebook and news website data as shown in Figs. 7(b) and (c) respectively.

Precision, recall and accuracy were computed for both the matrices and the results are shown in Table 1.

(a)

Actual/Prediction	0	0.1	0.2	0.3	0.4	0.5	0.6
0	2	2	0	0	0	0	0
0.1	1	7	1	0	1	0	0
0.2	0	1	5	2	0	0	0
0.3	0	1	1	4	1	2	1
0.4	0	2	2	1	4	1	1
0.5	0	0	0	2	2	3	2
0.6	0	1	1	1	1	1	6
0.7	0	0	0	0	0	1	1
0.8	0	0	0	0	0	0	0

(b)

Actual/Prediction	0	0.1	0.2	0.3	0.4
0	0	2	0	0	0
0.1	0	8	1	0	0
0.2	0	1	4	2	2
0.3	0	2	2	3	2
0.4	0	2	1	1	5
0.5	0	0	0	2	1
0.6	0	0	0	1	1
0.7	0	1	0	0	0
0.8	0	0	0	0	0

(c)

Actual/Prediction	0	0.1	0.2	0.3	0.4	0.5	0.6
0	0	2	0	0	0	0	0
0.1	0	3	1	1	0	0	0
0.2	0	3	6	0	2	1	0
0.3	0	2	1	5	1	1	1
0.4	0	1	2	1	4	1	1
0.5	0	0	0	1	1	4	0
0.6	0	0	0	0	0	2	2
0.7	0	0	0	0	1	1	1
0.8	0	0	0	0	0	0	1

Fig. 7. (a) Confusion matrix for twitter data. (b) Confusion matrix for Facebook data. (c) Confusion matrix for news website data

Table 1. .

Datasets used	Variables computed		
	Precision	Recall	Accuracy
Twitter dataset	0.56	0.79	0.86
Facebook dataset	0.65	0.73	0.87
News sites dataset	0.40	0.62	0.79

5 Result

In the present study, 36,500 tweets from twitter, 36,500 comments from Facebook and 36,500 votes of people from news sites were collected with an average of 100 per day for one year i.e. April 2017 to March 2018 which were related to 'iphoneX' using Python. The data was then scaled using min-max approach and neural networks were applied using R. The data was segmented into train and test dataset in the ratio 80:20 respectively. The actual and predicted values were used to define a confusion matrix. Precision, recall and accuracy of the result was calculated using this matrix. The Table 1 shows the values for all three datasets used in the present study.

6 Conclusion

From the above study we infer that Artificial Neural Networks can prove to be a better medium to perform sentiment analysis to a large amount of dataset as the accuracy obtained is between the range 79–87% and we do not need any extra amount of space to store the intermediate datasets as is the case with traditional methods used so far. Also the time consumed in training and testing of the data is also very less.

Acknowledgments. We thank open data platforms like twitter, Facebook and different news sites, which provide data to students like us to perform several types of studies and infer results from them. Also, we would like to thank free software providers like Anaconda platform, which is used in this study, to help people across the data science community to code, develop and analyse different problems.

References

1. Yazhini Priyanka, D., Senthilkumar, R.: Sampling techniques for streaming dataset using sentiment analysis. In: Fifth International Conference on Recent Trends in Information Technology(ICRTIT), pp: 1–6 (2016)
2. Trupthi, M., Pabboju, S., Narasimha, G.: Sentiment analysis on twitter using streaming API. In: 2017 IEEE 7th International Advance Computing Conference, pp. 915–919 (2017)
3. Ray, P., Chakrabarti, A.: Twitter sentiment analysis for product review using lexicon method. In: 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), pp. 211–216 (2017)
4. Fang, B., Liang, S., Zou, Q., Huang, W.: Research on sentiment analysis of financial texts based on web. In: 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC), pp. 248–252 (2017)
5. Soni, A.K.: Multi-lingual sentiment analysis of twitter data by using classification algorithms. In: Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), pp. 1–5 (2017)
6. Bhuiyan, H., Ara, J., Bardhan, R., Rashedul Islam, Md.: Retrieving youtube video by sentiment analysis on user comment. In: IEEE International Conference on Signal and Image Processing Applications (IEEE ICSIPA 2017), pp. 474–478. Malaysia (2017)

7. Likhari, M., Kasar, S.L.: Sentiment analysis using sentence minimization with natural language generation (NLG). In: 1st International Conference on Intelligent Systems and Information Management (ICISIM), pp. 134–140 (2017)
8. Ul Hassan, A., Hussain, J., Hussain, M., Sadiq, M., Lee, S.: Sentiment analysis of social networking sites (SNS) data using machine learning approach for the measurement of depression. In: International Conference on Information and Communication Technology Convergence (ICTC), pp. 138–140 (2017)
9. Prasad, A.G., Sanjana, S., Bhat, S.M., Harish, B.S.: Sentiment analysis for sarcasm detection on streaming short text data. In: 2nd International Conference on Knowledge Engineering and Applications, pp. 1–5 (2017)
10. Singla, Z., Randhawa, S., Jain, S.: Statistical and sentiment analysis of consumer product reviews. In: 8th ICCCNT, pp. 1–6 (2017)
11. Juneja, P., Ojha, U.: Casting online votes: to predict offline results using sentiment analysis by machine learning classifiers. In: 8th ICCCNT, pp. 1–6 (2017)
12. Sindhu, C., Vyas, D.V., Pradyoth, K.: Sentiment analysis based product rating using textual reviews. In: International Conference on Electronics, Communication and Aerospace Technology ICECA, vol. 2, pp. 727–731 (2017)
13. Wang, H., Wang, N., Yeung, D.-Y.: Collaborative deep learning for recommender systems, KDD '15, pp. 1235–1244 (2015)
14. Borele, P., Borikar, D.A.: An approach to sentiment analysis using artificial neural network with comparative analysis of different techniques. IOSR J. Comput. Eng. (IOSR-JCE) **18**(2). e-ISSN: 2278-0661, p-ISSN: 2278-8727
15. Image 1[a] source: https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html
16. Image 1[b] source: <https://inst.eecs.berkeley.edu/~cs182/sp07/assignments/a3-bp.html>