Sentiment Analysis of Product Reviews using Deep Learning

Conference Paper · September 2018

DOI: 10.1109/ICACCI.2018.8554551

Jagadeesh Panthati
Birla Institute of Technology and Science Pilani

Sentiment Analysis of Product Reviews using Deep Learning

Panthati Jagadeesh[1], Ranga Tarun Kumar, Challa Manish Reddy, Jasmine T.Bhaskar[2]
Amrita School of Engineering, Amrita Vishwa Vidyapeetham
Department of Computer Science & Engineering
[1]pjnani12@gmail.com, [2]jasmine@am.amrita.edu
Amritapuri, Kerala-690525, India.

***Abstract*—Sentiment analysis is one of the fastest growing research area, which helps customers to make better-informed purchase decisions through proper understanding and analysis of collective sentiments from the web and social media. It also provides organizations the ability to measure the impact of their social marketing strategies by identifying the public emotions towards the product or the events associated to them. Most of the studies done so far have focused on obtaining sentiment features by analyzing syntactic and lexical features that were explicitly expressed through sentiment words, emoticons and other special symbols. In this paper, we propose an approach to carry out the sentiment analysis of product reviews using deep learning. Unlike traditional machine learning methods, deep learning models do not depend on feature extractors as these features are learned directly during the training process. The main idea in this work is to use word2vec to learn word embedding and convolution neural networks to train and classify the sentiment classes of the product reviews. This combined word2vec-CNN model can be used to predict the sentiment of new product reviews.**

In this paper test for sentiment classification method with product reviews of mobile phones gathered from Amazon and show that our method gives better prediction accuracy than most of the existing methods.

***Keywords*—NLP(natural language processing), Sentiment analysis, CNN(convolutional neural network), Deep learning, machine learning, Text classification.**

I. Introduction

The rapid increase in the rate of internet users day by day leads to evolution of e-commerce and social media sites like Amazon, Flipkart, Facebook, Twitter etc. In 2017, 1.66 billion individuals acquired products from shopping sites and income added up to 2.3 trillion U.S dollars, by 2021 it might double. Nowadays reviews and rating have become an important source of information for consumers. Sentiment analysis is one of the major research type of NLP(Natural Language Processing) for tracking the opinion towards a particular product as positive or negative. It studies people's sentiments towards certain entities, like hotels, airlines, online shopping, Business Process Outsourcing (BPO) organization etc, highly increasing the number of customers for e-business made an impact on product reviews. Reviews are useful in making decisions for both the customer and manufacturer [1][2].

The opinion/sentiment is a state of mind, thought, or judgment provoked by a sentiment of the author(customer who is writing a review)[3].

Now, the first thing a person does when he or she wants to buy a product, is to see the kind of reviews and opinions that people have written on social media such as Facebook, Twitter and various other blogs or product review sites. Almost 95 percent of the customers consult customer reviews prior to making a purchase decision[4]. This has paved way for new research areas like sentiment analysis or opinion mining. Sentiment analysis provides insight to businesses by giving them immediate feedback on products, and measuring the impact of their social marketing strategies. This helps the manufacturer to identify new opportunities and manage their reputations[5].

Neural networks, which make use of an architecture inspired by the neurons in the human brain, are capable of solving almost any machine leaning classification problems. CNN (Convolution Neural Networks) gives impressive results for image classification[6]. In this work, we propose to use CNN (convolution neural networks deep-learning algorithm) [7][8] for sentiment analysis to get a better accuracy. In traditional machine learning algorithms like SVM[9], Naive Bayes etc, we need to extract features: the words or phrases that strongly express the opinion as positive or negative. However, Deep learning models do not depend on feature extractors because of the features learned during training process of algorithm. Deep learning has the potential to overcome many of the challenges faced by sentiment analysis[10].

The text reviews need to be converted to numeric data as CNN accepts only numeric input. To convert text to numeric data there are different approaches like one hot vector, TFIDF, GloVe, fast text(developed by Facebook)and Word2vec (Tomas Mikolov)[11]. In this work we propose to use word2vec to convert text into vectors or word embedding words expressing similar sentiment to have similar vector representation.

The organization of the rest of the paper is as follows: Section- 2 comprises of Related works, Solution approach in Section 3 and Experiment Results in Section 4 and finally, the Concluding remarks in Section-5.

II. Related Work

The model proposed in this paper draws inspiration from the existing works in sentiment analysis and deep learning. Though many researches have been carried out on sentiment analysis, most of them have used traditional machine learning classifiers like Naive Bayes, SVM etc. There are mainly three levels of approaches in sentiment analysis: word level, sentence level and document level. The word-level Sentiment analysis explores the orientation of the words in the review and their effect on the review. It uses a dictionary to give polarity to the words and calculates the overall polarity of the review to get the output[12]. Sentence level sentiment analysis considers whether a sentence in a document has a positive or negative sentiment by calculating overall polarity of the sentence. Document-level considers the whole document as single opinion (i.e., positive or negative).

Bo Pang Thumbs up?: had done sentiment classification on IMDB movie reviews dataset using various machine learning techniques[1][13] like SVM, maximum entropy classification,naive bayes and concluded the factors challenging sentiment analysis.

Houshmand Shirani-Mehr[8] had done a survey on various deep learning algorithms for semantic analysis on Stanford sentiment treebank as dataset using recurrent neural networks, recursive neural networks, and convolutional neural networks were implemented on the dataset and the results are compared. Aliaksei Sever yn [14] used an unsupervised neural model to train word embedding on twitter sentiment analysis organized by Semeval-2015 and that is further tuned by deep learning model and supervised the training data. Deep learning models have achieved amazing results in computer vision [6] and speech recognition. Alex Krizhevsky had done research on imagenet using deep learning algorithm (CNN), trained 1.2 million images to classify the imagenet contest dataset into 1000 different classes. Convolutional Neural Network has been proved very effective in solving the tasks related to computer vision. CNN contains the convolution layer, pooling layer and other layers, for complex problems the breadth and depth of CNN will continue to increase which would become limited by computing resources. we have virtual machines and GPU systems to train this type of models. Zhao Jianquiang presented a model called Glove-DCNN [15] for tweet sentiment classification and the experimental results indicate using pretrained word vectors together with deep learning method leads to better classification performance.

Deep learning model does not depend on feature extractors because features are learned during training process[16], whereas traditional models should do feature extraction independently. Features mean the words that emphatically express the opinion as positive or negative[17].

III. Proposed System

In this work, we propose to do sentiment analysis using a model which is a combination of word2vec and convolutional neural networks. At first we use word2vec to get the word embedding , which can be then used to construct the sentence vectors. These sentence vectors are then given as the input to CNN classifier. The framework of the proposed model is shown in Figure.1.

A. Pre-processing

The product reviews are collected from the Amazon web site and the punctuation's (like comma, full stop, semi colons), brackets, hyphens, numbers etc. are removed from the sentences. Every review in the dataset is then converted to lower case[18] and converted into tokens.

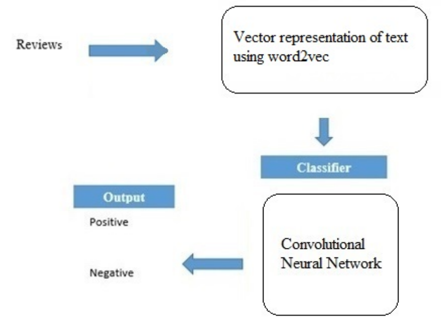

Figure 1: Framework of proposed model.

B. Word to vector conversion using Word2vec

After preprocessing the reviews are converted to vectors since CNN can understand only numeric data. In this model Word2vec is used to convert the text to word embeddings as the use of pretrained word embeddings improve the performance of the deep learning model. Word2vec, a tool developed by Google, is made of two algorithms namely[19] *Skip gram model and Continuous bag of words*.

As training Word2vec take a significantly long time, one can make use of pre-prepared models on the web like "GoogleNewsvectors-negative300.bin.gz", "English Wikipedia (Feb 2015) 1000 dimension-No stemming-10skipgram" and "German Wikipedia (Feb 2015) 300 dimension-No stemming-10cbow", to work with text. In this paper, Google's pre-trained word2vec is used to convert each word to a 300-dimensional vector. This model is trained on 100 billion words from Google News dataset and consists of 3 million word vectors. The word2vec pre-trained model can be downloaded from the[link].

Figure 2: Word2vec, relation between synonyms of "good" and "bad" keywords[20].

The different words, which often used in the same context (i.e. synonyms), will be very close in these vector representation [fig:2], which will be useful for our model to reduce the error rates.

The benefit of word2vec is, if we calculate differences between the vectors of words queen and king and differences between the vectors of words woman and man, we will find that these operations are similar to each other.
e.g "King – Man + Woman = ?" the result "Queen"[fig:3]

Figure 3: Word2vec, graphical representation of relationship between words[20].

There are various methods for implementing this embedding approach other than Word2vec are Glove, doc2vec, FastText etc. To derive vectors for reviews, we have different methods like we can calculate mean or the sum of words vectors or we can utilize unique models like Doc2vec. Training of Word2vec can take a significantly long time, and if you want to work with text, you can discover helpful pre-prepared models on the web (i.e., "GoogleNews-vectors-negative300.bin.gz", "English Wikipedia (Feb 2015) 1000 dimension-No stemming-10skipgram" and "German Wikipedia (Feb 2015) 300 dimension-No stemming-10cbow").

Note: Word2vec is utilized to get the vector representation of words and connection between them(reflect the distance of words) [21], this leads to initialize the parameters at a good point of CNN.

C. Formation of sentence matrix

In order to create the sentence feature vector which is the input to CNN, each token in the review is first mapped to the corresponding word embedding generated by the word2vec model. After mapping, each review is converted into a feature vector by concatenating the word embeddings. In order to have the sentence matrix of uniform dimension, smaller reviews are zero padded at the end. Then each of the review vectors are concatenated with the corresponding polarity vector to form the input sentence matrix to CNN.

D. Convolutional Neural Networks

Here we propose a convolutional neural network model to classify the sentiment of reviews as positive or negative. In deep learning, CNN is a feed-forward artificial neural network system which use a variety of multilayer perceptrons, that are intended to utilize negligible measures of preprocessing [wiki]. CNN has many fewer associations, made up of neurons that have learn-able weights and biases and they are easier to train. CNN is a powerful deep learning model which understands image content and imagenet classification is done with impressive results[6]. These results made CNN a popular tool in classification problems. A typical CNN architecture is shown in figure 5. A CNN comprises an input layer, multiple hidden layers and an output layer. Convolutional layers, activation functions, fully connected layers, pooling layers and other layers are part of hidden layers.

1) Convolutional layer: Convolutional layers apply a convolution activity to the input, passing the outcome to the following layer. A high number of neurons would be important, even in a shallow (inverse of deep) architecture, because of the substantial information associated with the size of the review (no. of words). A fully connected layer sends its yield to every neuron in the following layer whereas convolution operation conveys an answer for this issue as it diminishes the number of free parameters, enabling the system to be more profound with lesser parameters.

2) Max-pooling layer: Max-pooling uses the maximum value from a group of the parameter and reduces the number of parameters and computations in the network, it is a form of non-linear down-sampling.

3) Embedded layer: It is defined as the primary hidden layer of a network system, it is instated with arbitrary weights and will learn an embedding for all of the words in the training dataset and furthermore makes a look-up table for the embeddings.

4) Rectified linear unit: ReLU is an activation function to an output [range:(0 to ∞)]. It is the most popular activation function for deep leaning[fig:4].

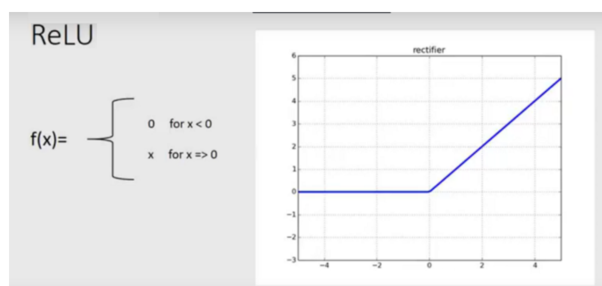

Figure 4: Graphical representation of ReLU[20]

5) Softmax: The softmax function is used in final layer which gives values in the range (0, 1). Mathematical equation for softmax function[14].

$$f_{\rm i}(x) = \frac{e^{x_{\rm i}}}{\sum_{j=1}^{J} e^{x_{\rm j}}} \tag{1}$$

6) Dropout: Increasing the number of epochs will have an over-fitted model, dropout is a technique used to tackle overfitting, it takes a float range[0 to 1], which is the fraction of the neurons to drop.

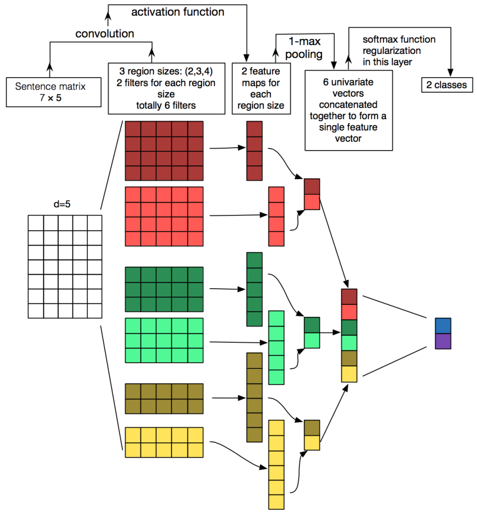

Figure 5: Convolutional neural network architecture[20].

7) Dense: Dense layer is a fully connected layer, represents a matrix vector multiplication. The value in the matrix are the trainable parameters which gets updated during back-propagation.

$$u^T.W, W \in R^{nxm} \tag{2}$$

8) Adadelta: Adadelta is an adaptive learning rate method.

In this paper, we have used keras open-source library and tensorflow as back-end for implementing CNN for sentiment analysis.

9) KERAS: Keras[22] is an open-source library intended to simplify the making of new deep learning models [23]. This high level neural network API can be run on top of deep learning frameworks like TensorFlow, Theano, etc. Known for its user friendliness and modularity, the tool is ideal for fast prototyping. The tool is optimized for both CPU and GPU.

10) TensorFlow : TensorFlow is an open-source library for high performance numerical computation. This framework developed by the Google Brain team can be used to create artificial neural network for deep learning. Its flexible architecture allows easy deployment of computation across a variety of platforms like CPUs, GPUs and TPUs.

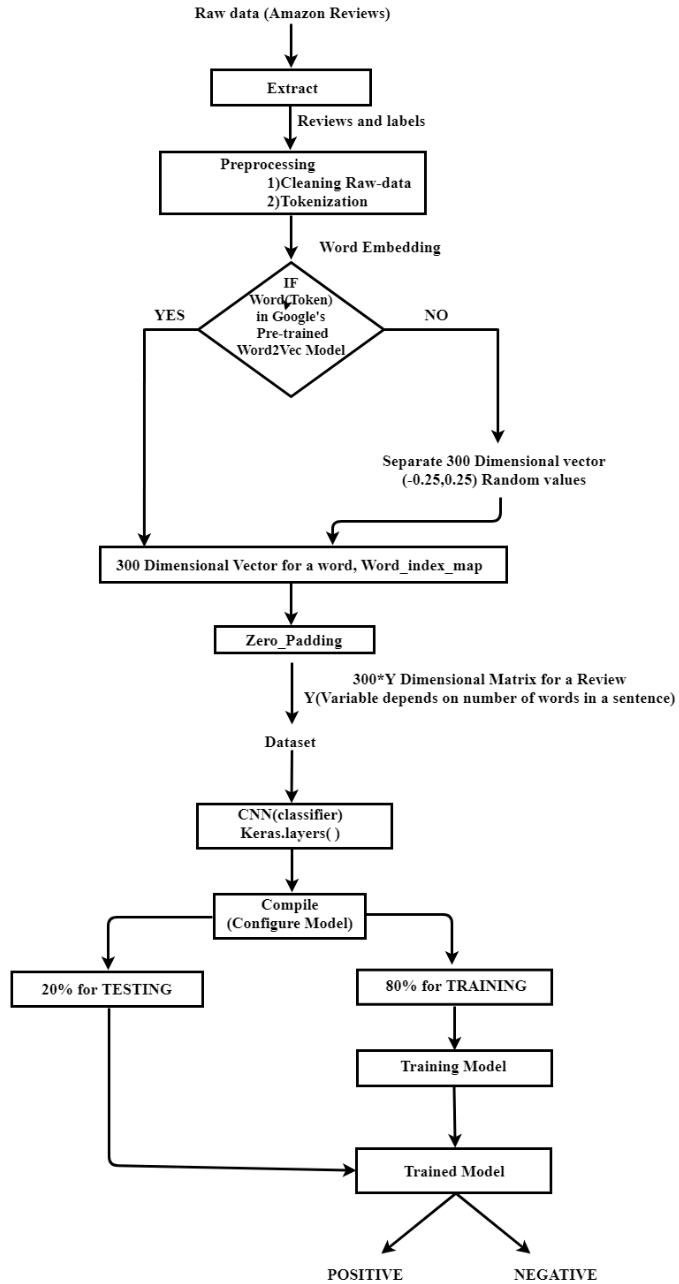

Figure 6: Design

IV. Experiments and Evaluation

A. Dataset

This project uses the static dataset downloaded from[24] [link]. Dataset is constructed with 1000 reviews with labels(1 as positive and 0 as negative). The idea is to evaluate the performance of our proposed CNN architecture on Amazon product reviews(mobile phone reviews), it includes 500 positive and 500 negative reviews. To make the evaluation process more precise 5761 reviews are chosen from 4,00,000 reviews released by amazon website[link], where customers gave 1 to 5 rating for their reviews, this project uses reviews rated as 1 (0-polarity) and reviews rated as 5 (1- polarity). The idea is to evaluate the performance of our proposed Word2vec-CNN architecture on Amazon product reviews(mobile phone reviews).

B. Experiment

At first,the reviews and labels are extracted from the text file and different preprocessing steps like cleaning the string, stopword removal, numerics removal etc are performed. After tokenising the sentence, each token(word) is converted into the 300-dimensional vector using Word2vec (google's pre-trained Word2vec model). For the words that are not present in the pre-trained model, a separate word vector is created by randomly choosing the values between -0.25 and 0.25 so that the unknown vectors have (approx) the same variance as pre-trained ones. All these 300-dimensional vectors are stored in an array. Since the length of the different sentences(reviews) vary from each other zero padding is done for the shorter sentences(reviews) in order to make the size of input vectors equal. As the maximum number of words/tokens in a review in the data set is 800 , a sentence matrix of size 800 x 300 is formed.

The obtained sentence matrix , concatenated with the polarity (label of each review) is then fed as input to the CNN(convolutional neural network). CNN architecture in this work consists of 1)Convolutional layer, having kernel size 8 and 300 feature maps(outputs of convolutional layer) followed by 2) ReLU activation 3) Max-pooling 4)Dropout 5) Dense, fully connected layer, and Softmax activation function to get values in the range range(0 to 1) . An optimizer adadelta is used for setting adaptive learning rate. Since the CNN may not learn the best weights after I epoch, we propose to do 3 epochs in this experiment. The best weights are defined as the weights that minimize the loss function and hence multiple passes (epochs) on the training dataset is required . If the number of epochs are high , it may lead to over fitting. From the data set 80 percent of the data is used for training and the rest 20 percent is used for testing the model.

C. Result

The evaluation metrics used in this experiments are accuracy, precision, recall etc. The accuracy of the model is calculated using various methods like finding validation accuracy using the best threshold value t, finding validation AUC(area under the curve), ROC curve which plots the graph between TPR(true positive rate) and FPR(false positive rate) etc.
Figure 9 shows the confusion matrix for the proposed model. When the model is trained with 779 reviews from a set of 1118 reviews and tested with the remaining 219 reviews, it correctly classified 188 reviews showing an accuracy of 85.844 percent. But when the size of the data set is increased and trained the model with 4643 reviews out of 5763 reviews and tested with 1118 reviews, the model correctly classified 1014 reviews and the accuracy increased to 90.697 percent. The accuracy of the model is calculated as

$$\begin{aligned} Accuracy &= \frac{correctlyidentifiedreviews}{totalno.ofreviews} \\ &= \frac{TP+TN}{TP+TN+FP+FN} \end{aligned} \tag{3}$$

Confusion matrix shows the number of reviews that are correctly classified and misclassified .The entries in the confusion matrix are
TP(True positives): No. of positive reviews that are correctly labelled by the classifier.
TN(True negatives): No. of negative reviews that are correctly labelled by the classifier.
FP(False positives): No. of negative reviews that are incorrectly labelled as positive by the classifier.
FN(False negatives): No. of positive reviews that are mislabeled as negative by the classifier.
ROC curve is a graph plotting tpr against fpr at various threshold values. It is a natural way to an analysis of decision

	Predicted: NEGATIVE	Predicted: POSITIVE	
Actual: NEGATIVE	TN = 586	FP = 59	645
Actual: POSITIVE	FN = 45	TP = 428	473
	631	487	**1118**

Figure 7: Confusion matrix for 5761 reviews(testing- 1118)

making. The nearer the ROC curve is to the upper left corner, the higher the overall accuracy of the test[25].
True-positive rate is also called as sensitivity, recall(ratio of

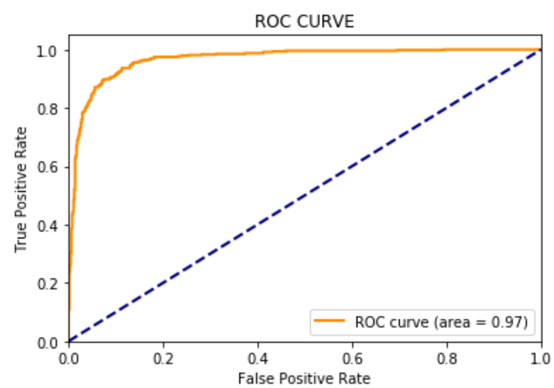

Figure 8: Output diagram for ROC Curve for 5761 Reviews[26].

no. of correctly identified positives and total no. of positives).

$$Recall = TPR = \frac{TP}{(TP+FN)} \tag{4}$$

False-positive rate is also called as the fall-out,1-specificity (ratio of no. of incorrectly identified negatives and total no. of negatives).

$$FPR = \frac{FP}{(FP+TN)} \tag{5}$$

$$Precision = \frac{TP}{(TP+FP)} \tag{6}$$

$$AveragePrecision = \sum_n (R_n - R_{n-1}) P_n$$

where $P_n and R_n$ are the precision and recall at the nth threshold. A pair (R_k, P_k) is referred to as an operating point.

To evaluate the performance of the proposed Word2vec+CNN model, it is also compared with other

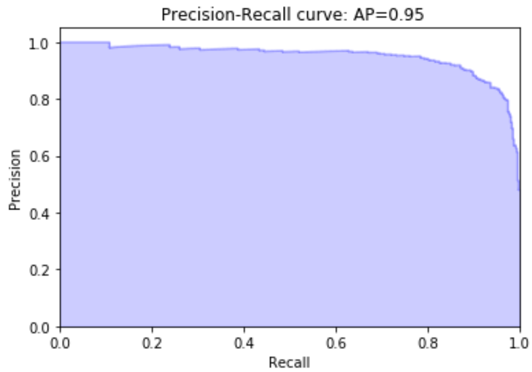

Figure 9: Output diagram for precision recall curve for 5761 Reviews[26].

traditional machine learning models like Naive Bayes and SVM. The accuracy of these models are listed below.

Model	Acc
Bigram+NaiveBayes	0.55
BOW+NaiveBayes	0.678
NormalizedBigram+NB	0.525
SVM	0.71
Word2Vec+CNN	0.8442

Table I: Result for 1000 Reviews

Model	Acc
Bigram+NaiveBayes	0.441
BOW+NaiveBayes	0.735
NormalizedBigram+NB	0.587
SVM	0.74
Word2Vec+CNN	0.91323

Table II: Result for 5761 Reviews

In order to create the naive bayes classifiers, after pre-processing, bigrams are constructed from the data set. Then a bigram matrix is constructed using the probabilities of bigrams. For 1000 reviews it forms a matrix of [1000*((max(number of words in a review)-1))] .We have used gaussianNB() which is one of the naive-bayes algorithms for classification and calculated the accuracy of the models. For the BOW +Naive Bayes model, a bag-of-words matrix is formed with the tokens and to normalize the matrix or to avoid zeros, Truncated SVD() is used. For all tokens in the dataset it forms a matrix of [total tokens * unigrams].

V. Conclusion

Sentiment Analysis is an NLP task to identify a particular product review to be positive or negative. Neural networks and deep learning are becoming popular in solving almost any machine learning classification problem. In this paper, we have proposed a model for sentiment analysis of product reviews using word2vec and CNN. Google's pre-trained word2vec model is used to convert the text to word embeddings. CNN in this model is implemented using the open source deep learning frameworks Keras and Tensorflow. Our experiment results show that the proposed approach entitles a better accuracy compared to the existing traditional machine learning models and the accuracy of the model increases with a substantial increase in the size of the dataset.

In this work, we have used a static dataset, but in the future fetching of reviews dynamically from E-Commerce websites or blogs can be done. For production or service based organisations, it is not enough to know whether a review is positive or negative, they would like to identify which aspects of their products or services are positive or negative. So this work can be extended to identify these aspects from the reviews. Due to the limitation of computational power and the high computation time required we had to limit the size of the data set in this experiment, but in future it can be carried out on large scale data using GPU.[]

References

[1] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics, 2002.

[2] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media*, pages 30–38. Association for Computational Linguistics, 2011.

[3] Jasmine Bhaskar, K Sruthi, and Prema Nedungadi. Enhanced sentiment analysis of informal textual communication in social media by considering objective words and intensifiers. In *Recent Advances and Innovations in Engineering (ICRAIE), 2014*, pages 1–6. IEEE, 2014.

[4] Christine. The importance of sentiment analysis in social media analysis, 2015.

[5] Meena Rambocas, João Gama, et al. Marketing research: The role of sentiment analysis. Technical report, Universidade do Porto, Faculdade de Economia do Porto, 2013.

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[7] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[8] Houshmand Shirani-Mehr. Applications of deep learning to sentiment analysis of movie reviews. Technical report, Technical report, Stanford University, 2014.

[9] Jasmine Bhaskar, K Sruthi, and Prema Nedungadi. Hybrid approach for emotion classification of audio conversation based on text and speech mining. *Procedia Computer Science*, 46:635–643, 2015.

[10] Prerana Singhal and Pushpak Bhattacharyya. Sentiment analysis and deep learning: a survey, 2016.

[11] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, L Sutskever, and G Zweig. word2vec. *Google Scholar*, 2014.

[12] Nikos Engonopoulos, Angeliki Lazaridou, Georgios Paliouras, and Konstantinos Chandrinos. Els: a word-level method for entity-level sentiment analysis. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, page 12. ACM, 2011.

[13] Bo Pang, Lillian Lee, et al. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135, 2008.

[14] Aliaksei Severyn and Alessandro Moschitti. Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 959–962. ACM, 2015.

[15] Zhao Jianqiang, Gui Xiaolin, and Zhang Xuejun. Deep convolution neural networks for twitter sentiment analysis. *IEEE Access*, 6:23253–23260, 2018.

[16] Huy Nguyen and Minh-Le Nguyen. A deep neural architecture for sentence-level sentiment classification in twitter social networking. *arXiv preprint arXiv:1706.08032*, 2017.

[17] Minqing Hu and Bing Liu. Mining opinion features in customer reviews. In *AAAI*, volume 4, pages 755–760, 2004.

[18] Ewan Klein Steven Bird and Edward Loper. Natural language toolkit: http://www.nltk.org/home.

[19] Xin Rong. word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*, 2014.

[20] Google image search.

[21] David Meyer. How exactly does word2vec work? 2016.

[22] François Chollet et al. Keras. https://keras.io, 2015.

[23] Guoyong Cai and Binbin Xia. Convolutional neural networks for multimedia sentiment analysis. In *Natural Language Processing and Chinese Computing*, pages 159–167. Springer, 2015.

[24] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

[25] Mark H Zweig and Gregory Campbell. Receiver-operating characteristic (roc) plots: a fundamental evaluation tool in clinical medicine. *Clinical chemistry*, 39(4):561–577, 1993.

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[27] Cicero dos Santos and Maira Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78, 2014.

[28] Emma Haddi, Xiaohui Liu, and Yong Shi. The role of text preprocessing in sentiment analysis. *Procedia Computer Science*, 17:26–32, 2013.

[29] Xi Ouyang, Pan Zhou, Cheng Hua Li, and Lijun Liu. Sentiment analysis using convolutional neural network. In *Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on*, pages 2359–2364. IEEE, 2015.

[30] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[31] Global retail e-commerce sales 2014-2021 — statistic.