Interpretation of Product Quality by Rule based and Machine Learning Approaches Using Opinion Mining

Madhapur Vineelaswathi and Chuan-Ming Liu

July 23, 2021

Interpretation of Product Quality by Rule based and Machine Learning Approaches using Opinion Mining

[1]*M Vineelaswathi*, [1*]*Chuan-Ming Liu*
[1]Dept. of Electrical Engineering and Computer Science, National Taipei University of Technology.
[1]t108998406@ntut.org.tw,
[1]*Dept. of Computer Science and Information Eng, National Taipei University of Technology.
*Corresponding author: cmliu@ntut.edu.tw

Abstract

It is no longer important to have direct interaction with people to know their opinion regarding the product they are using. Everything is available online in commercial and social networking platforms. To analyze the emotion of the customer or a public individual based on the tweet or review to find out how he thinks the product is a difficult task. In this work, we detect the emotion /opinion of a consumer with respect to a particular product, by using different methodologies and algorithms and analyze which is the most optimal solution for unstructured data. The main aim of this paper is 1) To test the traditional topic models like LDA and LSA by VADER algorithm which is a Rule Based approach and 2) To test the Machine Learning techniques like CNN-LSTM and Bi-LSTM and obtain the best suitable model in terms of loss, accuracy and also to find a model with less training and testing time i.e., operating time for the given unstructured data.

Keywords: Opinion Mining, Topic Modelling Techniques, VADER, CNN with LSTM and Bi-LSTM

1. Introduction

Opinion Mining which is also referred as Sentiment Analysis has a great impact in the business which deals with user/public opinions for analyzing the emotions, feelings, sentiments, appraisals, attitude and gaining knowledge related to organizations, issues, particular topic or product or an occasion. In this research we use Opinion Mining for commercial purposes such as product assessments; it systematically models the customers' needs and perspectives, resulting in consumer perspectives and benefits. Some applications related to recommender systems, stock predictions, business intelligence applications, product marketing, and managing the reputation of the companies [1], [2].

Sentiment analysis helps the researchers' or Data scientists' to widely research about the public opinion, brand's reputation, market research and emotions of the customers. To interpret the product quality these steps are to be followed: 1) Tokenize the complete document into single words or phrases. 2) Analyze the sentiment of each sentence of that particular word. 3) Assign a score based on the positivity and negativity of each word or phrase. 4) Assess the quality of the product. Using lexicon dictionaries plays a vital role in obtaining positivity and negativity. Those dictionaries consist of adjective phrases, noun phrases and even some adverb phrases [12]. Manually evaluating the sentiment scores has lot of difficulties because the sentiment scores for word 'sad' can be assigned as '0.5' by an individual and '-0.75' by the other. So maintaining lexicon dictionaries constantly are important. In this work we have not only take noun phrases but also noun-adjective and nun-adverb phrases also. The interpretation of product quality comes with many limitations by Rule based approach and also with LSTM. So we try to use the supervised techniques like CNN-LSTM and Bi-LSTM to overcome the limitations.

2. Literature Review

Over recent years, a considerable number of studies have been focused on Aspect-based Opinion Mining. It has a huge amount of potential for predicting a person's mood based on an event or his feelings about a product based on his tweet or review. It is extremely difficult for a computer to comprehend what a human is feeling. The majority of data generated these days are unstructured data which doesn't fit neatly into a structure or a framework. Since my Sentiment Analysis is based on unsupervised, I will mention some of the unsupervised techniques to Opinion Mining. The technique mentioned in [3] is one of the first approaches regarding Aspect Based Opinion Mining. The authors proposed an unsupervised approach to do Sentiment Analysis. Before this paper, there used to be only identifying topics and then assigning a label of opinion to the particular extracted feature. So the authors in [3] have proposed an approach in which they can find out the aspect with respect to sentiment simultaneously.

I started my study from work presented by [4] in which they explained various lexicon techniques mainly by VADER. In [2] the authors gave a detailed survey on all the existing techniques and approaches in relation with Opinion Mining and Sentiment Analysis for retrieval of information. This work gave a detailed explanation of the bag of words and how positive, negative or neutral using these classifiers. An example of a perfect unsupervised approach is given by [5] in which they extract an aspect which matches the aspect you search automatically.

Now-a-days there is a trend to do a lexicon based opinion mining for different languages like Chinese, Persian, Japanese rather than doing it for English. The authors in [6] not only developed a Language Independent Aspect Based Opinion Mining but also they found out the polarity and subjectivity with the effect of intensifiers by not only using noun and adjective dictionaries but they also used adverb dictionaries in extracting the exact opinion of the review which was very helpful in my study.

The most commonly used approaches focused on subject modeling techniques are Latent Semantic analysis [7], and Latent Dirichlet Allocation [8]. These experiments are unsupervised and only use a well-known subject modeling technique to classify the topic. A wide variety of experiments were conducted based on [8] which helped in evolving these techniques. On the other hand, some of the recent studies concentrated on supervised machine learning methods like in [9] they have shown how the aspect extraction effects when they use n-gram classifiers for feature extraction.

[10] Offers a comprehensive overview of current sentiment analysis approaches. They also explore the summary of sentence-level sentiment analysis and analyze and deploy based on benchmarks of labeled datasets in which the data set is protected by product and feedback shared on social networking websites. [11] Uses an extension of LSTM which performs better than simple LSTM. They also conducted experiments with adaptive recursive neural networks. There was also a massive improvement when compared with feature-based SVM.

The target-dependent classification with lexicon enhanced neural networks also has poorer results than the approach used in [11]. [12] Used supporting evidence from [11] to compare a labeled dataset with a rule-based approach using VADER and a supervised technique called Long Short Term Memory (LSTM), and discovered that the supervised technique using LSTM outperforms the Rule-based Lexicon approach using VADER. They have selected LSTM because it works fair enough for the data which are sequential in nature and works better than Recurrent Neural Network (RNN) from which LSTM is extended. With the inspiration from [6], [7], [8] and [12] I have extended my studies.

3. Methodology and Aspect Detection

Layers in the Proposed Model: The Aspect Detection and performance model is proposed using Bidirectional LSTM and CNN-LSTM networks. The architecture of the proposed model is given in Figure 1 has an aspect extraction layer and a recurrent layer for Bi-LSTM model whereas Aspect extraction layer, Convolutional layer and a recurrent layer for CNN-LSTM. As shown in Figure 1, the aspect detection layer is the first layer of the proposed model. The reviews written by the customers' needs to be processed and converted into aspects by using different topic modelling techniques and also can be done by rule based approach for better understanding. To extract the aspect by topic modelling techniques, we use LDA [8] and LSA [7]. The LDA helps to keep the optimal model with few parameters. In LDA, we use sampling techniques to improve the efficiency of the matrices. The most commonly used sampling technique in LDA is Gibbs Sampling. The crucial part in LDA is, it works backwards. For E.g. Let us assume that we have 'a' number of documents and by using LDA, the topics are extracted with 't' in number where we may have each sentence is divided into 't' topics. LDA does the reverse engineering process from the document to identify the topics that are detected in the complete dataset. In LSA, the statistical approach of the related document is represented in each row of the matrix U_k (document-term matrix). These vectors have a length of 'k', which is the number of selected topics. The matrix V_k is a feature vector for the terms in our data (term-topic matrix). At the end, after applying SVD we obtain vectors for each term and each document with a length of a topic vector 'k'. With these matrices we can also apply a cosine similarity algorithm to find out the similarity between terms, topics and documents. The aspect detection is also done by one more rule based approach using 'Text Blob'.

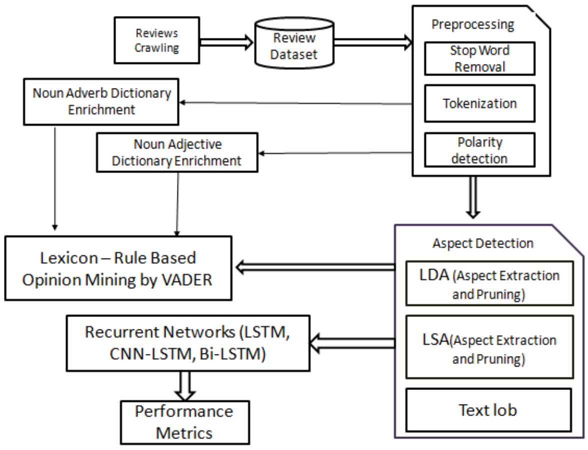

Figure 1: Architecture of Proposed Model

Data Crawling: To train and evaluate the proposed Bi-LSTM model, we have to prepare an offline dataset. Even if there are few research outputs for this system, as far as researchers' understandings there is no direct dataset accessible for the products till the date. Due to this, we have to web crawl from the promising website named "AMAZON " that contains good truth about the electronic products. The web crawling is done with the help of the 'Beautiful Soup' package. Web crawler is nothing but a bot which is used to crawl the required data from the online sites. Web crawling is nothing but which helps the company or researcher to grab all the data into a library so that anyone can make us of it if it is publically available. The people who need to make use of those files will just read the overview and to find out what it's about in order to help categorize and classify the library's volumes by topic.

From the 'Amazon' website we select three products i.e., OnePlus 8 Pro Ultramarine Blue, 5G Unlocked Android Smartphone U.S Version, 12GB RAM+256GB Storage, 120Hz Fluid Display, Quad Camera, Wireless Charge, Sony a7 III ILCE 7M3/B Full-Frame Mirror less Interchangeable-Lens Camera and Apple MacBook Air (13-inch Retina Display, 8GB RAM, 256GB SSD Storage) for three countries USA, India and UK (United Kingdom). The product quality interpretation is done for three products for three countries mentioned above. By using the topic modelling techniques like LDA and LSA we detect the aspects and interpret the data according to the aspects extracted.

Data Preprocessing: The crawled dataset needs some preprocessing and cleaning. The first step in data preprocessing is tokenization. Tokenization is nothing but dividing the complete sentence / paragraph of a corpus into smaller ones i.e., a sentence will be divided into many single words. The NLTK package is used for tokenization. While we tokenize the sentence/ paragraph into words we import the 'sent_tokenize' function from the 'nltk' package. This function inside 'nltk' takes a corpus/ paragraph and applies it with a lot of regular expressions, it also takes all sentences into a particular list and converts it into sentences/ words respectively.

The next step is lemmatization. Lemmatization is the process of reducing infected words to the word stem in a meaningful way. For E.g. There are some words like' history', 'historical', 'historically' then after applying lemmatization for the sentences where we have these words, the three words will be combined as one word 'history' in a meaningful way. We chose lemmatization over stemming because the problem with stemming is that it produces an immediate representation of the word which may or may not have the meaning. The function used for lemmatization is 'WordNetLemmatizer ()'.

The third step is stop word removal, which helps in removing stop words like is, was, the, and. For removal of the stopwords we first need to download the stopwords from nltk package. The command 'set' helps us to take the unique stopwords in English. The set of unique words are compared with the other words in the dataset. If the words in the dataset are not in the list of stopwords then stemming can be performed on the words. This method is called list comprehension.

The latter step is data cleaning. While cleaning, first we must convert all the uppercase letters into lowercase ones because the model considers 'quality' and 'Quality' as two different words. To avoid that confusion between the same words we must change all the letters into lower case. The other special characters like numbers, URL's, \t, \n need to be eliminated. Later, we split up the words and lemmatize it for each word in the review and remove the stopwords.

After preprocessing and cleaning, we then detect the polarity or sentiment score for the whole dataset. Polarity may vary from -1 to +1. '-1' is considered as an extremely negative polarity whereas '+1' is considered as an extremely positive polarity. We can obtain the sentiment score of the complete dataset which varies from -1 to +1.

Aspect Detection: The feature extraction is done either the way, i.e., features are extracted either by using topic modelling techniques (by LDA, LSA) or by using 'Text Blob'. Before extracting the aspect, small cleaning needs to be done i.e., pruning. In pruning, we try to do Aspect Pruning and Compactness Pruning.

Compactness pruning- Count the no. of small words and words without an English definition, only if 'nonsensical' or short words do not make up more than 40% of the phrase then we add it to the cleaned list, effectively pruning the ones that are not added.

Redundancy pruning- Find the common words among all phrases, if the size of matched phrases set is smaller than 30% of the cleaned phrases, then consider the phrase as non-redundant.

First step here is to check out the noun phrases from the single strings, which will be used for frequent feature extraction. Frequency of features are obtained, the count of each feature which are obtained after redundancy pruning are collected. Here comes the most important step to set the threshold value. I have selected the frequent feature threshold as (max_count)/100. If the feature extracted > threshold value, then it is considered as the most frequently used word in the dataset. The Aspect extraction is done by two traditional classifiers LDA, LSA and 'Text Blob'. After the features are extracted we apply VADER. To obtain the features based on the opinion and also to interpret the product quality we must detect the features as mentioned in the previous step, with reference to [6] the opinion words are nothing but the features detected from the documents. The features are extracted based on two dictionaries i.e., "Noun Adverb" and "Noun Adjectives" which helps in finding the opinion words. The predominance of the intensifiers play a major role which gives the knowledge of actual emotion involved to the company. For E.g. the reviews like "focus of the camera lens is bad" and "focus of the camera lens are too bad" is definitely different from each other. The models need to understand the difference between these two and helps in finding the correct polarity with strength involved in each sentence. These factors are needed to be considered while extracting the aspects.

VADER stands for Valence Aware Dictionary for Sentiment Reasoning is a model used for text sentiment analysis that is sensitive to both polarity (positive/negative) and intensity (strength) of emotion and also used to accommodate the sentiments or emotions expressed in various platforms. It is available in the NLTK package and can be applied directly to unlabeled text data. VADER sentiment is a subpart of Lexicon based approach which is also a part of Rule based approach [13].In this study we mainly concentrate on another term called Subjectivity along with Polarity and Intensity [13]. For E.g. VADER helps in differentiating the importance of the words. In other words, with the help of intensity the polarity of words ‘good’ is a bit different from ‘great’. Any other models can tell that both ‘good’ and ‘great’ are positive words but with the help of VADER we can have the polarity scores as ‘0.65’ and ‘0.8’ respectively.

A word or text’s sentiment score can be calculated by including the strength of each word in a sentence along with the subjectivity. VADER has an ability not only to understand whether the particular word is positive or negative but it can also tell how positive and how negative the word is in a given sentence.

$$Normalizedscore = \frac{score}{\sqrt{(score * score) + alpha}} \tag{1}$$

With the aid of the SentimentIntensityAnalyzer() method, they offer another score in addition to positive, negative and neutral scores. It is named as compound scores. These compound scores are the normalized scores which vary from -1(the extreme negative value) and +1(the extreme positive value) [12], [14]. From the above Equation (1) the value of **score** is nothing but the sum of all the polarity scores of all the aspects from a sentence whereas the **alpha** value is a constant value mostly 15.

Word Embedding by GloVe: The more and more research interest in NLP and Opinion Mining proves that by deep learning models than the machine learning techniques and also the research had a turning point when the researchers converted words to vectors by using word embedding techniques. The word embedding techniques are mainly of two types – Word2Vec, GloVe. Word2Vec word embedding technique is a toolkit used to produce word vectors by Google in 2013, which simply and efficiently solves the problem of word embedding and gives input for LSTM, CNN-LSTM, and Bi-LSTM.

In this research, we use GloVe [15] embedding technique which works on the principle of nearest neighbors. It finds out the Euclidean distance or cosine similarity between the two or more word vectors and finds out the semantic relation or similarity between the words and assigns the value for a vector. We use the Glove file (GloVe.6B.100d.txt) which has 40,000 words with the dimension of 100. The training of the Glove model is done with non-zero entries of the term-term co-occurrence matrix, which tells us how frequent the word occurred in the document or a review. GloVe is a log-bilinear model with least-square objectives.

Rule Based and Machine Learning Approaches: In Rule based approach, after data collection, preprocessing & cleaning the topics or aspects extracted by topic modelling techniques is given as input to the VADER algorithm. The VADER algorithm generates a score in terms of percentage that reflects the portion of text which have the same sentiment group. In VADER lexicon when a topic/ aspect extracted by either topic modelling technique are given to the model, it results in giving Sentiment scores. The polarity score lies in between -1 to +1.

By using Rule based approach, the best optimal solution is chosen in between the two traditional topic modelling techniques i.e., LDA and LSA. At last, by using logistic regression we find the accuracy and performance metrics.

In the Machine Learning approach, data collection, preprocessing & cleaning the topic or aspects extracted by ‘LDA’ in the matrix form are given as input to the word embedding model. The word embedding model used here is GloVe technique. The vectors obtained from GloVe technique in the matrix form are given as input to the embedding layer for the deep learning models like CNN-LSTM and Bi-LSTM. Thus we can find the best optimal solution among CNN-LSTM and Bi-LSTM in terms of losses and accuracy and also find out the performance of models with the metrics like specificity, sensitivity, FPR and overall accuracy. At last, we find out the True Positives, True Negatives, and False Positives & False Negatives and show it in ROC curve.

4. Experimental Results and Discussions

To get the optimal model that has high accuracy, several experiments are performed using collected databases on the different network architectures of the proposed model. Using Machine learning techniques, we performed three experiments on different datasets of different countries by choosing the best suitable hyperparameters, hidden layers. In the case of CNN-LSTM model, we chose the best number of Convolutional and max pooling layers. Besides, after each operation, we used batch normalization, ReLu and sigmoid activation functions. The arrangement of the two experiments are as follows: i) CNN-LSTM contains one Embedding layer, one Convolutional layer, a max pooling layer and LSTM layer, ii) Bi-LSTM consists of one Embedding layer, a forward LSTM layer and a backward LSTM layer. Finally, the optimal model is chosen and several optimizers, including Adam, Adamax, Adadelta and AdaGrad are tested on the selected optimal model.

On the other hand, to get the optimal result in MLP, we perform several experiments with a certain number of hidden layers and the optimizer functions. To see the effects of the number of hidden layers in MLP, we also have three experiments with different parameters such as the number of neurons, normalization functions and activation functions as well as optimizers.

Each optimizer has its own set of parameters. In our experiments, the optimizer parameters are kept at their default values. The rectified linear activation function is used for the entire experiments in the first dense layer to mitigate the vanishing gradient descent problem. The sums of the squares of the difference between the actual and predicted values are calculated to estimate the loss of the proposed method. Each network is trained with 40 epochs and batch-size of 100.

Results for Rule Based Approach:

After preprocessing and data cleaning is done, first the aspects are detected by using the ‘text blob’ module which helps in telling us the positivity and negativity of the aspect extracted. The negatives are marked as ‘0’ and positives are marked as ‘1’. The positivity and negativity of the particular aspect is determined based on the review of the product given by the customer. The Figure 2 shows the aspects extracted by the 'text blob' package for Sony digital camera of the USA.

	review	label
0	superior focus system	1
1	great video	1
2	good low light	1
3	low noise	0
4	gorgeous camera	1

Figure 2: Aspects Extracted by 'text blob' module

From the Figure 2 we can assess that the features like "focus of the camera" is given as '1', which means this feature of the Sony Camera is considered to be positive and similarly the other features like 'great video' has a positive feedback and 'noise' has a negative feedback i.e., the camera has a lot of noise.

	0	neg	neu	pos	compound
0	phone zero	0.000	1.000	0.000	0.0000
1	low surroundings	0.677	0.323	0.000	-0.2732
2	brilliant let	0.000	0.208	0.792	0.5859
3	phone good	0.000	0.256	0.744	0.4404
4	op delivery	0.000	1.000	0.000	0.0000
5	hey case	0.000	1.000	0.000	0.0000
6	lock chargers	0.000	1.000	0.000	0.0000

Figure 3: Features by LSA for OnePlus_UK

The aspects extracted by 'LSA' and' LDA' are shown in Figure 3 and Figure 4 respectively. These extracted features are for the OnePlus mobile and for the United Kingdom country. It shows the top 7 extracted features by topic modelling techniques with the positive, negative, neutral and compound scores. In LSA, unwanted and non-semantic aspects are extracted like 'op delivery' and 'hey case'.

	0	neg	neu	pos	compound
0	phone screen	0.0	1.000	0.000	0.0000
1	screen oneplus	0.0	1.000	0.000	0.0000
2	oneplus battery	0.0	1.000	0.000	0.0000
3	battery great	0.0	0.196	0.804	0.6249
4	great fast	0.0	0.196	0.804	0.6249
5	fast good	0.0	0.256	0.744	0.4404
6	good camera	0.0	0.256	0.744	0.4404

Figure 4: Features by LDA for OnePlus_UK

As shown in Figure 3 the features like "phone good" and "brilliant let" have a positive score whereas "low surroundings" has negative score. In Figure 4 the features like "battery great" and "great fast" has a positive score which says that it has great battery life and works faster than other mobiles.

From the above Figure 5 we can say that the extracted feature 'great battery' has a compound score of nearly 0.98, which means the feature battery, is reviewed as extremely positive in most of the reviews. Similarly, it goes the same for the other features like 'good camera', 'fast good' which can be considered as having an opinion of extremely positive review.

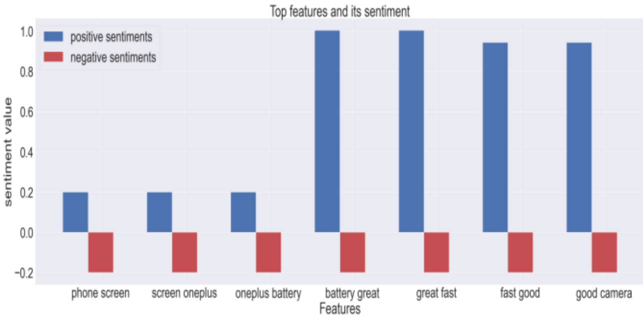

Figure 5: Top Features with its Sentiment for OnePlus_UK by LDA

Results for Machine Learning Approach:

The actual data for Machine Learning approaches is aspects obtained from 'LDA' which is considered as accurate among topic modelling techniques. Aspects detected by 'LDA' for One Plus mobile are shown in Figure 4. With CNN-LSTM and Bi-LSTM they converge after the 20th and 15th epoch respectively. In Figure 6 training and testing accuracy and losses of the machine learning model CNN-LSTM are shown. As a result, the accuracy for training and testing are increased to 89.82% and 85.04% respectively. The training and testing losses are decreased to 0.09 and 0.13 respectively.

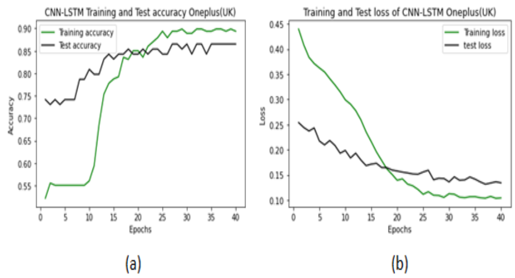

Figure 6: (a) Training and Testing accuracy and (b) Losses of CNN-LSTM

In Figure 7 training and testing accuracy and losses of the machine learning model Bi-LSTM are shown. As a result, the accuracy for training and testing are increased to 87.32% and 82.01% respectively. The training and testing losses are decreased to 0.11 and 0.16 respectively.

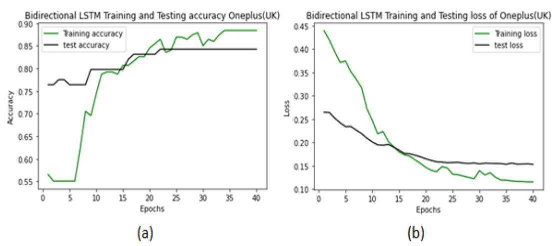

Figure 7: (a) Training and Testing accuracy and (b) Losses of Bi-LSTM

CNN-LSTM converges faster than the other two models i.e., Bi-LSTM and Simple LSTM with high detection accuracy and losses are also less compared to other two as shown in Figure 8.

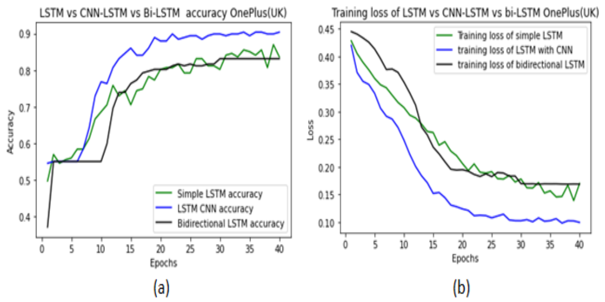

Figure 8: (a) Validation accuracy and (b) Loss of CNN-LSTM, Bi-LSTM and LSTM

In machine learning a model is trained and the loss of prediction is computed using the ground truth and predicted output. Based on the loss, the network parameters of the model must be tweaked using a function called optimizer which can minimize the loss of the model. There are different optimization algorithms, some of these are Stochastic gradient descent (SGD), Adaptive Gradient descent (AdaGrad), AdaDelta, and Adaptive moment (Adam).

In addition to the Adam optimizer used in model selection, the other four optimizers such as SGD, Adamax, AdaGrad and AdaDelta are checked on the selected model (CNN-LSTM). As shown in Table 1, Adam optimizer has better aspect detection accuracy of 89.82% on CNN-LSTM. ON the other hand, SGD shows the minimum aspect detection accuracy of 88.44% and the effect of optimizers are shown in Figure 9.

Model	SGD	Adamax	AdaDelta	AdaGrad	Adam
CNN-LSTM	88.44	89.19	88.67	89.80	89.82
Bi-LSTM	85.19	86.54	85.23	86.99	87.32

Table 1: Effects of Optimizers in Aspect detection of CNN-LSTM and Bi-LSTM

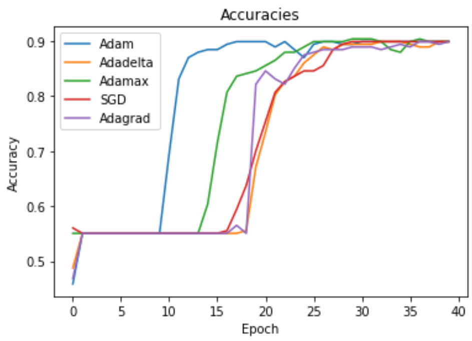

Figure 9: Effect of Optimizers.

ROC curve is nothing but receiver operating characteristic, which shows the performance of the classified models at different thresholds. In this research we are proposing two different models i.e., one with CNN-LSTM and the other with Bi-LSTM, we try to find the performance of these models in comparison with the previous models. This ROC curve is a plot between True Positive Rate (TPR) which is also called Recall or Sensitivity and False Positive Rate (FPR) which is also known as 1-Specificity.

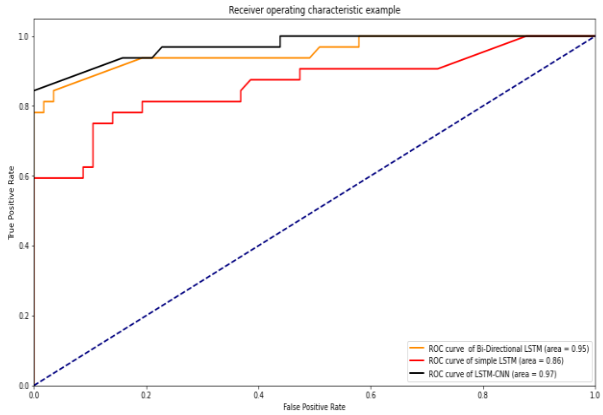

Figure 10: ROC Curve for all the models

The plot shown in Figure 10 is the plot at different thresholds. The area under the ROC curve (AUC) gives the dimensional space under the ROC curve which ranges from (0, 0) to (1, 1). AUC is a composite measure of performance that takes into account all potential categorization criteria. The chance that the model rates a random positive example higher than a random negative example is one method to analyze AUC. The Area under the Curve value varies from 0 to 1. The AUC curve tells us that the model, whose predictions are ‘true’ for all the values has the AUC value of ‘1’ and the one with all the predictions as ‘False’ has AUC as ‘0’. In our model AUC for LSTM is 0.86, Bi-LSTM is 0.95, and CNN-LSTM is 0.97.

5. Conclusions and Future Work

In this research, we investigate several deep learning methods to find out whether the extracted positive or negative aspect is actually detected as positive or negative. To compare the results of the proposed method, previously proposed models are analyzed. For the proposed model, appropriate training and testing dataset is prepared. Our experiment is to interpret the product quality and also compare the Rule based approach and Machine learning approach.

In the first approach, the VADER algorithm is used to find out the positivity and negativity in the extracted aspects. After the data preprocessing, cleaning and stop word removal is done we extract the aspects by ‘text blob’ and also by using the traditional topic modelling techniques i.e., LSA and LDA which is the actual data. While detecting the aspects we also perform aspect pruning and compactness pruning. After pruning is done, we calculate the threshold value where we can find the top frequent aspects mentioned by the customers who bought that product. Then the VADER algorithm is applied for these extracted aspects and finds how positive and how negative our aspect of the product is. As mentioned in the above chapters we get the compound scores or normalized scores which vary from -1 to +1 and tell us how well the features are working after the usage or the first impression that customers had. In this first approach, the aspects extracted by LDA show a better semantic relationship than LSA and 'text blob'.

In the latter approach, i.e., Machine Learning approach, we perform two experiments: the first one is by using Bi-LSTM and the second one is CNN-LSTM, which are proposed to detect the aspects. In the first experiment after cleaning and preprocessing is done features are extracted and apply word embedding technique. Like the first approach the aspects detected by the 'LDA' are considered. While detecting the aspects we also perform aspect pruning and compactness pruning. After pruning is done, we calculate the threshold value where we can find the top frequent aspects mentioned by the customers who bought that product. Later, we perform word embedding technique, GloVe and try to convert the word into a vector and given as input Bi-LSTM layers and analyzes the performance of this proposed model.

In the second experiment, we perform all the things that are mentioned above for the first experiment. We give the embedding matrix obtained from word embedding technique as input to the other model i.e., CNN-LSTM and investigate the performance of this model. According to the results, CNN-LSTM is chosen as the optimal model for detecting the aspects. Besides the optimizers, both models are evaluated. Later, different optimizer were applied for the optimal model (CNN-LSTM) and also Bi-LSTM, and found that the best suitable optimizer for detection is 'Adam' optimizer. This approach shows that our proposed methods show better results in detection of the aspect when compared to previously proposed models.

In this research, successful aspect detection with Machine Learning approach is presented. The motivation behind the study was to get rid of the weakness of the existing systems. In order to get rid of these weaknesses, it is first necessary to undertake a thorough analysis on the current literature. After that, a complete solution that eliminates many of the drawbacks of the existing methods is proposed. According to the experience gained so far from this study, we get a satisfactory result for both Bi-LSTM and CNN-LSTM. In future, incorporating more training data and using language models which will improve the detection of the aspects independent of the language.

References

[1]Liu, B., 2012. "Sentiment analysis and opinion mining". Synthesis lectures on human language technologies, 5(1), pp.1-167.

[2]Bo Pang and Lillian Lee., 2008, "Opinion Mining and Sentiment Analysis", Foundations and Trends in Information Retrieval: Vol. 2: No. 1–2, pp 1-135.

[3]Lin, C. and He, Y., 2009, November. "Joint sentiment/topic model for sentiment analysis." In Proceedings of the 18th ACM conference on Information and knowledge management (pp. 375-384).

[4]Hutto, C. and Gilbert, E., 2014, May. "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In Proceedings of the International AAAI Conference on Web and Social Media (Vol. 8, No. 1).

[5]Akhtar, N., Zubair, N., Kumar, A. and Ahmad, T., 2017. "Aspect based sentiment oriented summarization of hotel reviews". Procedia computer science, 115, pp.563-571.

[6]Hosseinzadeh Bendarkheili, F., Mohammadi Baghmolaei, R. and Ahmadi, A., 2019, April. "Product Quality Assessment using Opinion Mining in Persian Online Shopping". In 2019 27th Iranian Conference on Electrical Engineering (ICEE) (pp. 1917-1921). IEEE.

[7]Landauer, T.K., Foltz, P.W. and Laham, D., 1998. An introduction to latent semantic analysis. Discourse processes, 25(2-3), pp.259-284.

[8]Blei, D.M., Ng, A.Y. and Jordan, M.I., 2003. Latent dirichlet allocation. The Journal of machine Learning research, 3, pp.993-1022.

[9]Tripathy, A., Agrawal, A. and Rath, S.K., 2016. Classification of sentiment reviews using n-gram machine learning approach. Expert Systems with Applications, 57, pp.117-126.

[10]Ribeiro, Filipe N., et al. "Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods", EPJ Data Science 5.1 (2016): 1-29

[11]Tang, D., Qin, B., Feng, X. and Liu, T., 2015. Effective LSTMs for target-dependent sentiment classification. arXiv preprint arXiv:1512.01100.

[12]Adarsh, R., Patil, A., Rayar, S. and Veena, K.M., 2019. Comparison of VADER and LSTM for sentiment analysis. International Journal of Recent Technology and Engineering, 7(6), pp.540-543.

[13]Towards Data Science. VADER algorithms for sentiment analysis. Accessed 2021-05-11 [Online].Available:https://towardsdatascience.com/senti mental-analysis-using-vader-a3415fef7664

[14]Amin, A., Hossain, I., Akther, A. and Alam, K.M., 2019, February. Bengali vader: A sentiment analysis approach using modified vader. In 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE) (pp. 1-6). IEEE.

[15]Pennington, J., Socher, R. and Manning, C.D., 2014, October. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).