

Implementation of Deep Learning Based Sentiment Classification and Product Aspect Analysis.

Shweta Dhabekar¹, Dr. M. D. Patil²

¹M.E Student, Ramrao Adik Institute of Technology, D. Y. Patil Deemed to be University, Navi Mumbai, India.

²Professor, Ramrao Adik Institute of Technology, D. Y. Patil Deemed to be University, Navi Mumbai, India.

Abstract. With the increase in E-Commerce businesses in the last decade, the sentiment analysis of product reviews has gained a lot of attention in linguistic research. In literature, the survey depicts the majority of the research done emphasizes on mere polarity identification of the reviews. The proposed system emphasized on classifying the sentiment polarity and the product aspect identification from the reviews. Proposed work experimented with traditional machine learning techniques as well as deep neural networks such as Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Long Short Term Memory (LSTM) Networks. The proposed system gives a better understanding of these algorithms by comparing the outcomes. The Deep Learning approach in the proposed work successfully provides a mechanism which identifies the review polarity and intensity of the reviews and also analyses the short form words used by people in the reviews. The experimental results in this work, applied on amazon product dataset, shows that the LSTM model works the best for sentiment analysis and intensity of reviews with 93% accuracy. This research work also predicts polarity for short-form word reviews which is the common trend these days while writing the reviews.

1 Introduction

In recent years, there has been an increase in research activities focused on analyzing sentiment in textual resources. The number of articles published on sentiment analysis has been growing in recent years. Sentiment analysis or opinion mining is one of the subtopics of this research, in which it computationally investigates people's views, assessments, attitudes and emotions regarding entities, individuals, situations, events, texts.

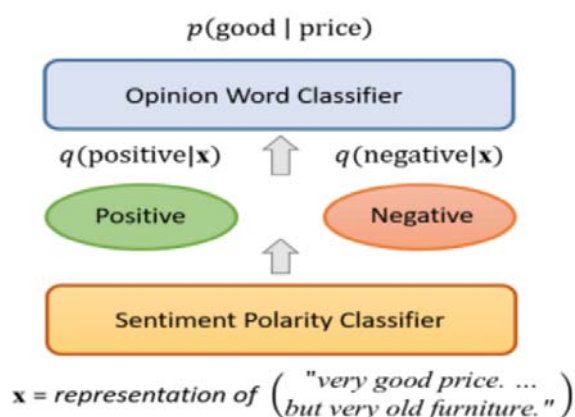


Fig. 1. Research carried out on sentiment analysis.

This approach has a wide range of applications. E-Commerce Businesses, for example, are constantly interested in consumer thoughts or opinions about their products and services. Before using a service or purchasing a product, most of the consumers are keen to know what other people think about it. Lastly, research scholars [2] utilize this data to conduct semantic analysis of trends in market and buyers sentiment, which might improve forecasting product quality. Despite this, monitoring online opinion sites, as well as extracting the information hidden in them, is still a difficult undertaking due to the proliferation of different sites. In extensive forum postings and blogs, each site often has a large amount of opinionated content that is not always easy to read. The ordinary reader may face trouble locating appropriate websites and precisely extracting the hidden details and viewpoints included therein [5]. Furthermore, teaching a computer to understand sarcasm is a difficult and time-consuming endeavour, considering that

* Corresponding author: author@e-mail.org

computers cannot yet think like humans. The goal of this work is to identify positive and negative opinions of various products and to develop a model which is a supervised model, having the capability to polarize enormous volumes of data. Customers' reviews and ratings, obtained from Amazon's Consumer Reviews, are used as a dataset for the proposed system. In proposed research, it retrieves the features from the input dataset and uses those features to build various supervised models. Traditional machine learning mechanisms and deep learning techniques like RNN, CNN and LSTM are included in these models. The accuracy for various such models were analyzed and a thorough evaluation of the obtained sentiments regarding the reviews was done. These models make use of 70% of the data for training and 30% of data serves as testing data.

Yi-Fan and Maria Soledad Elli gathered sentiments by analyzing reviews and evaluated the results to develop a business model in their paper[3]. Authors said that this model gained them a high level of precision. The major classifiers were Support Vector Machine (SVM) and Multinomial Naive Bayesian (MNB). Callen Rain[6] advocated expanding current Natural Language Processing (NLP) research. To determine whether a review was good or not, the Naive Bayes classifier was used by authors. In the field of sentiment analysis, deep-learning neural networks are popular too. For the sentiment labeling challenge, some researchers deployed a convolutional neural network with the purpose of avoiding unnecessary task-specific feature engineering. In contrast to this, [8] recommended employing recursive neural networks to gain a better grip over sentiment prediction.

2 Related Work

There have been several research papers published so far on product reviews, sentiment analysis and opinion mining. Wei Zhao et. el. [1] proposed a deep learning approach for product review sentiment classification,

which uses existing ratings as input values. The work includes two steps: (1) to make the model understand the representation which evaluates the sentiment values for input sentences through ratings; (2) deploy the classification layer above the embedding layer and make the use of tagged sentences for supervised learning. Author proposes two types of low level network structures for determining review text, one is Convolutional Feature Extractors and the other is Long Short-Term Memory network. In [2], the author has given a deep survey of reviews and concluded that the internet is the most preferred method of learning, acquiring ideas and obtaining various product ratings. Millions of evaluations on a commodity, service, individual or location are posted on the internet daily. It's tough to read and evaluate such assessments one by one because of their sheer volume and quantity. As shown in Figure 2, there are three major types of sentiment classification based on access levels. If the classification is Document level, it's assumed that every single document is an opinion. If the classification is Sentiment Level, it checks if every sentence has some opinion expressed in the sentence. If the classification is Aspect level, it searches for the various aspects about which the sentiments are expressed in the opinion. Natural language processing (NLP), text analysis and determining the sentiment polarity of posted reviews are all part of the Sentiment analysis process. Its core objective is to recognize the sentiment behind the given review.

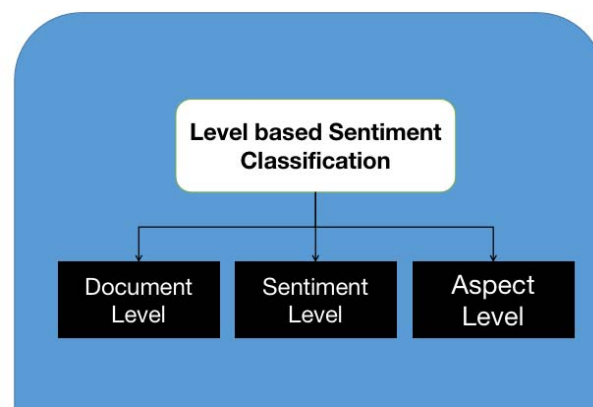


Fig. 2. Types of sentiment analysis.

Many algorithms exist to deal with NLP challenges. The categorization of reviews is presented in the proposed system. In comparison to Naive Bayes' and extreme entropy approaches, this research proves that SVM provides higher accuracy. On Yelp's rating dataset, Xu Yun[8] et al used current supervised learning techniques like supporting vector machine or perceptron algorithm to determine review's rating.

3 Proposed System

3.1 Proposed Work

Product review analysis is basically more concerned about analyzing the text to determine the polarity of the reviews. This is the most tedious task as the reviews are user-written and there is no specific pattern or rules to be mapped. As the data is highly temporal in nature, the use of artificial intelligence and right choice of AI model is presented in this research. Though traditional machine learning techniques like SVM behaved better in various NLP systems over the years, they have some limitations, which deep learning models easily overcome to a greater extent. Deep learning models have the advantage of requiring no hand-crafted characteristics to be carefully optimized. It takes word embeddings as input instead of features, which contain context information and the intermediate layers of the neural network learn the features during the training phase. Deep learning makes it possible to learn representations automatically. This eliminates the requirement for manual feature or word information construction. The proposed system comprises 3 major steps:

1. The system emphasizes on the polarity detection of the reviews.
2. The system emphasizes on also integrating the short form words used nowadays in reviews, which is not included in conventional data dictionaries.
3. The proposed system also emphasizes the polarity aspect of the product.

3.2 Proposed System Design

The proposed system first uses Meta Dictionary Mapping for short form words in reviews. Nowadays, there is a trend of people writing textual reviews by using short form words e.g. "gud" for "good" or "nyc" for "nice". So the standard dictionary does not have any mappings for such words and hence any review stating "the product is gud" won't be considered to be positive. Hence the proposed system maintains the key value pair of the short form words and actual words, thereby making the system capable of mapping the short form words as well for sentiment prediction.

The proposed system's overall flow is as follows:

- The proposed system first loads the dataset and cleans the dataset by removing unnecessary columns from the dataset.
- The cleaned dataset is then processed by VADER sentiment analysis after text pre-processing, for labeling the review's polarity (0,1).
- The Meta Dictionary is maintained for storing the short form words and their corresponding actual words, so that before the model is being trained, words are mapped to actual words.
- After labelling data for its polarity, the data is provided to the proposed LSTM model with some preliminary neural network configurations.
- The model training is then started. If the training output is not optimal, the configurations are varied and the model is retrained till the optimal results are obtained.
- Once the optimal results are obtained, the system then makes use of Part of Speech (POS) tagging on the input reviews. These input reviews are provided as test reviews for testing the model and product review aspects are extracted by segregating nouns from the tagged review tokens.

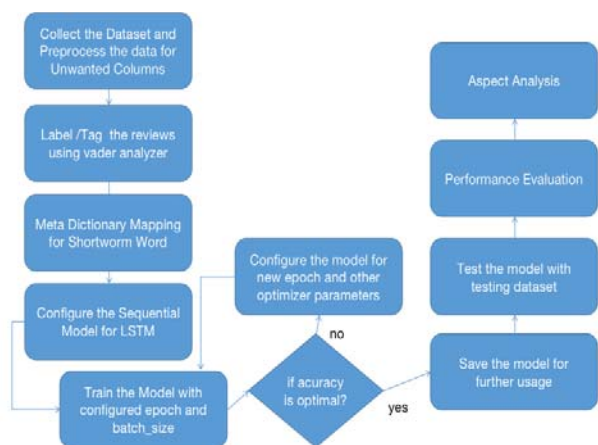


Fig. 3. Proposed system flow chart

3.2.1 Text Pre-Processing

Text Pre-Processing: To easily extract the features from the raw product reviews, this research first transforms our data into normalized unigrams. This research performs text pre-processing by removing all capitalization, tokenize the text by continuous sequences of alphabetic characters (remove whitespaces, symbols, punctuation and numbers). For each token, it filters out english stop words from the NLTK stopwords corpus. From this, it gives the summary statistics of the listed data such as total number of reviews, total reviews for different ratings values as seen in Figure 4.

3.2.2 Data Preparation

In order to avoid unwanted data, only some part of the entire data is used.

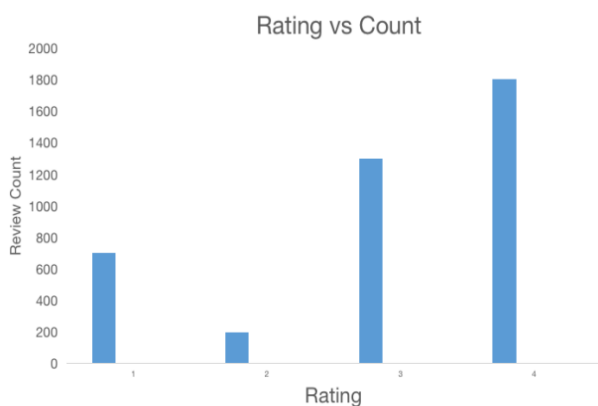


Fig. 4. Product rating v/s Count of Reviews

For example, reviews having a positive sentiment with ratings of 4 & 5 and reviews having a negative sentiment with ratings of 1 & 2 are considered, whereas reviews with neutral sentiments having a rating of 3 are dropped. The proposed systems' primary purpose is to analyze the reviews for its polarity whether it is positive or negative. There are two preliminary phases to this process. To convert a text into the corresponding numerical representation, it first locates a word embedding. Second, to identify whether proposed work makes use of machine learning methods or deep learning architectures to fit the numerical representations of text. Frequency-based embedding, such as the Bag of Words (BoW) model, is a typical technique for word embedding. The BoW model extracts a vocabulary list from a corpus and represents each text using various word counting approaches. This process depicts how well BoW works with supervised learning algorithms. The very first step in text pre-processing is to convert the raw reviews into cleaned reviews. This procedure entails eliminating non-character words such as numerals and symbols, removing stop words, converting all the review sentences to lower case and stemming words to obtain the root words. In Step 2, the CountVectorizer from the sklearn package is used to combine tokenization with occurrence counting in a single review. Tokenization in python is the process of splitting the entire text into smaller chunks or words. In Step 3, the proposed system converts the text in the review to the corresponding numerical representations (feature vectors) and fits these feature vectors to the supervised learning algorithm in Step 4. Tf - idf transform is used to scale down the impact of frequently appearing words. Tf stands for 'term frequency' and counts how many times a word has appeared in a single review. Df stands for 'document frequency and counts how many times a particular word has been used from a list of reviews.

Table 1. Classification Report of Random Forest Classifier.

Polarity	Precision	Recall	F1-Score
0(negative)	0.87	0.82	0.84
1(positive)	0.94	0.96	0.95
Average	0.92	0.92	0.92

3.2.3 Applying Machine Learning Algorithm

As per the previous research done, various algorithms in machine learning are used for categorical prediction of review sentiments. Among various ML algorithms, Random Forest Classifier (RFC) has given better performance. So, to check the performance of RFC on the dataset, RFC with base_estimator as ‘Decision Tree’ and n_estimators = ‘100’ was set to fit the model to achieve the prediction for review classification.

3.2.4 Applying Deep Learning LSTM

The proposed system makes use of the traditional Neural Network Mechanism for Creating the LSTM Network, but with the use of additional data pre-processing steps for cleaning the input dataset of reviews extracted from the site. There are time-stamps in LSTM, but the block of “MEMORY” in the LSTM cell is an additional feature. In general, LSTM cells consist of these building blocks 1. Input Gate “I”, Candidate layer “C”, Forget Gate “f”, Output Gate “O”, Hidden state “H” and a Memory state “C”. The diagram below shows the LSTM cell at time t. LSTM cell takes the old memory state C_{t-1} and performs unit wise multiplication particularly with forget gate (f)

$$C_t = C_{t-1} * f_t \quad (1)$$

If it's found that the forget gate is equal to 0, then the old stored value in the memory state is straight away

forgotten.

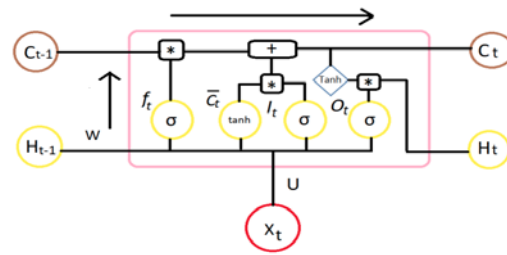


Fig. 5. LSTM Cell Overview

If it's found that the forget gate is equal to 1 then the old value in the memory state is straight away passed to the cell. So, with the present memory state C_t it calculates the new cell state in coordination with the C layer and the input state.

$$C_t = C_t + (I_t * C^*) \quad (2)$$

Where, C_t = present memory at time-stamp t which gets passed to next time-stamp. Fig 6. shows a flow diagram for C_t . The proposed system architecture has the standard Neural Network Blocks for generating the model for analyzing the sentiment from the reviews.

- 1) Long Short Term Memory networks are a special kind of Recurrent Neural Networks (RNN), capable of learning long-term dependencies.
- 2) Prepare X_{train} and X_{test} to 2D tensor and Train a simple LSTM
- 3) Load word embedding model which was trained in the above step.
- 4) Construct an embedding layer using an embedding matrix as weights and Train the LSTM model with embedding.
- 5) Compile and fit the model using the log loss function and ADAM optimizer.

The LSTM gate activation functions are often logistic functions. There are recurrent connections going in and coming out of the LSTM gates.

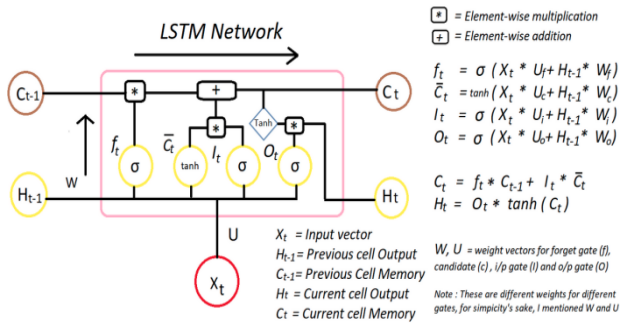


Fig. 6. LSTM Cell flow diagram

Table 2. Model History Report.

Model: "Sequential"		
Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 779, 128)	256800
lstm (LSTM)	(None, 196)	256800
dense (Dense)	(None, 2)	394
Total params: 511,194		
Trainable params: 511,194		
Non-trainable params: 0		
None		
(3355, 779) (3355, 2)		
(839, 779) (839, 2)		
Train on 2684 samples, validate on 671 samples		

Connection weights, which the model learns in the training phase, decides the operation of gates. Sequential model is created by using keras.models, the first layer which is an embedding layer is added to the sequential model with maximum features of data, 128 embedding dimensions and input data shape. The LSTM layer is added to the model with 196 units given to the layer for dimensionality of the output space. Dense layer is added with an activation function which is "softmax" which means that the neural network has n neurons and the activation function will be "softmax". Then the model is compiled by model.compile() which configures the model for training. The model is compiled with loss, optimizer and accuracy.

4 Results and Discussion

4.1 Experimental Setup and Dataset

The information includes Amazon reviews on mobile phones. The product feature names and the associated emotions are marked in the reviews. Clearly, each of these statements mention the entity being reviewed as well as the level of opinion held about it. After conducting a variety of experiments, this research concluded that the third party website reviews are not sufficient for the research purpose. Therefore, more than 5000 reviews were labelled manually. On a scale of -5 to 5, aspect ratings are labelled. This system has additionally pre-processed publicly available data, so as to fit the indicated element into corresponding buckets. By indicating value "1", it's seen that they are referring to portability with a positive attitude. This necessitated some word and sentiment translation manually. The dataset is a subset of a larger dataset of Amazon Product Reviews accessible on the Kaggle repository, as well as a synthetic version with certain changes (removal of some unnecessary columns from the original dataset). The proposed work presently will be predicting the intensity analysis of the review. The present model is able to determine the polarity of whether the obtained review is positive or negative. Also the proposed implementation consists of polarity intensity identification also, which will provide a more detailed prediction. e.g. "the product is good" - Positive, but "the product is extremely good" - Very positive. The Proposed work does not only determine the polarity of the reviews, but also determines the aspects of the reviews. Which means that if the review indicates some aspect of the product and the reason for its polarity, the proposed system will extract that information too e.g. "the product battery life is awesome", so the system will also determine that the review is positive and the aspect being "battery" of the product. The proposed system is also capable of identifying the polarity of reviews in which short form

words are used for expressing the sentiments. If the person writes the review “the product is gud”, the word “gud” is not a standard dictionary word, so the CNN model is deployed to predict the actual words from the trained dataset, which comprises of actual words and possible short form words that can be used in now a days. As seen in Fig 7, the input review uses the word “gud” which is a short form, but the final analysis shows the actual word “good” instead of the short form word.

```

Display is gud but battey is weak
Original 1
good
(0, 1903) 1.0
[2]
Original 1
weak
(0, 4848) 1.0
[0]
[['display', 'good', 'positive'], ['battey', 'weak', 'Negative']]
    
```

Fig. 7. Product Aspect Analysis

The meta dictionary is maintained for mapping short-form words with actual words which are replaced while the pre-processing phase is computed.

4.2 Performance Evaluation

The proposed system is evaluated based on various parameters such as accuracy, precision recall and F1 Score. The Confusion Matrix is the base for all these values as shown below. All these performance metrics are dependent on true positive, true negative, false positive and false negative values:

- a. True Positive TP: The expected value was positive and the predicted value was also Positive.
- b. True Negative TN: The expected value was negative and the predicted value was also Positive.
- c. False Positive FP: The expected value was negative and the predicted value was positive which is False prediction.
- d. False Negative FN: The expected value was positive and the predicted value was negative which is False prediction

Precision: Precision talks about how precise your model is out of those predicted positives, how many of them are actual positives, calculated as $Precision = TP/(TP+FP)$.

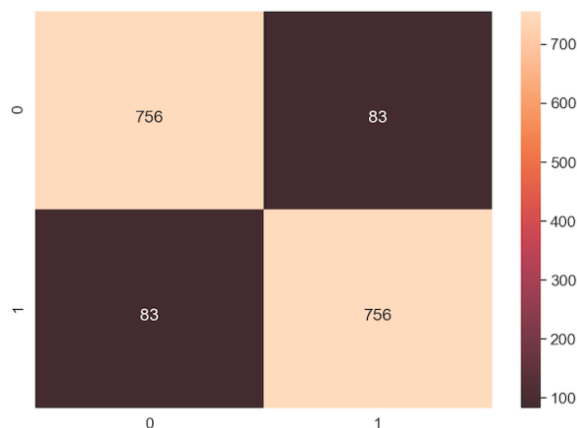


Fig. 8. Proposed Model Confusion Matrix

Table 3. Classification Report of proposed system.

Polarity	Precision	Recall	F1-Score
0(negative)	0.91	0.90	0.905
1(positive)	0.95	0.96	0.935
Average	0.93	0.93	0.92

Recall: Recall actually calculates how many of the Actual Positives our model captures through labeling it as Positive (True Positive),calculated as $Recall = TP/(TP+FN)$.

F1 Score: F1 Score might be a better measure to use if its needed to seek a balance between Precision and Recall AND there is an uneven class distribution (large number of Actual Negatives),calculated as $F1\ Score = 2 \times ((Precision * Recall)/(Precision + Recall))$.

5 Conclusion

The obtained results show that the efficiency of the proposed LSTM model is better than the traditional machine learning models. Also, the proposed system does not restrict its scope over polarity or sentiment

prediction, it digs into the polarity intensity such as whether the review is just positive or very positive and same for negative reviews. The proposed system results also show that the different aspects of the product about which positive is being said or negative is being said are also identified. The efficiency of the model is based on performance evaluation; it is seen that it outperforms Machine Learning model performance parameters. Hence it is observed that the proposed model gives more accurate and elaborative details about the reviews which is helpful in terms of analyzing the aspects of the products whose polarity is also identified. So because of this, consumers and service providers will get better clarification on products' market value and help them make important business decisions.

References

1. Wei Zhao, Ziyu Guan, Long Chen, Xiaofei He, Deng Cai, Beidou Wang, Quan Wang, Weakly-Supervised Deep Embedding for Product Review Sentiment Analysis, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 30, No. 1, January 2018.
2. T. K. Shivaprasad, Jyothi Shetty, "Sentiment analysis of product reviews: A review", 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), doi 10.1109/ICICCT.2017.7975207
3. Y. Fang, Hai Tan, Multi-Strategy Sentiment Analysis of Consumer Reviews Based on Semantic Fuzziness, *INSPEC Accession Number: 17745421, DOI: 10.1109/ACCESS.2018.2820025, IEEE, Page(s): 20625 - 20631.*
4. W. Songpan, The Analysis and Prediction of Customer Review Rating Using Opinion Mining, 2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA), *NSPEC Accession Number: 17010009, 10.1109/SERA.2017.7965709, IEEE.*
5. R. Abinaya, P. Aishwaryaa, S. Baavana, N.D. Thamarai Selvi, Automatic Sentiment Analysis of User Reviews, 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR), *NSPEC Accession Number: 16559951, 10.1109/TIAR.2016.7801231, IEEE*
6. M. Kavousi, Sepehr Saadatmand, Estimating the Rating of the Reviews Based on the Text, *First International Conference on Data Analytics & Learning 2018.*
7. S. Zirpe; Bela Joglekar, Polarity Shift Detection Approaches in Sentiment Analysis: A survey, 2017 International Conference on Inventive Systems and Control (ICISC), *INSPEC Accession Number: 17259793, DOI: 10.1109/ICISC.2017.8068737, IEEE*
8. M. Kulkarni, Mayuri Lingayat, Effective Product Ranking Method based on Opinion Mining, *June 2015 International Journal of Computer Applications 120(18):33-37, DOI:10.5120/21331-4306*
9. J. Fontana Rava; Gabriella Pasi; Marco Viviani, Feature Analysis for Fake Review Detection through Supervised Classification, 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA), *INSPEC Accession No.17521817, 10.1109/DSAA.2017.51, IEEE*
10. M. Chauhan, Divakar Yadav, Sentimental Analysis of Product Based Reviews Using Machine Learning Approaches, *Journal of Network Communications and Emerging Technologies (JNCET) Volume 5, Special Issue 2, December(2015).*
11. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research, 12(Aug):2493–2537, 2011.*
12. K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528. *ACM, 2003.*
13. M. S. Elli and Y.-F. Wang. Amazon reviews, business analytics with sentiment analysis.
14. S. Hota and S. Pathak. Knn classifier-based approach for multi-class sentiment analysis of twitter data. In *International Journal of Engineering Technology*, pages 1372–1375. *SPC, 2018.*
15. B. Liu and L. Zhang. A Survey of Opinion Mining and Sentiment Analysis, pages 415–463. *Springer US, Boston, MA, 2012.*
16. C. Rain. Sentiment analysis in amazon reviews using probabilistic machine learning. *Swarthmore College, 2013.*
17. R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. 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.
18. Bo Pang and Lillian Lee, Opinion Mining and Sentiment Analysis, Foundations and Trends in Information Retrieval, 2008.
 19. Theo M.V. Janssen, Frege, contextuality and compositionality, Computer Science, University of Amsterdam, 2002.
 20. Xin Wang, Yuanchao Liu, Chengjie Sun, Baoxun Wang and Xiaolong Wang, Predicting Polarities of Tweets by Composing Word Embeddings with Long Short-Term Memory, Harbin Institute of Technology, Harbin, China, 2015.
 21. Ziqian Zeng, Wenxuan Zhou, Xin Liu, Yangqiu Song, A Variational Approach to Weakly Supervised Document-Level Multi-Aspect Sentiment Classification, Computer Science, Computation and Language, 10 April 2019, arXiv:1904.05055.