

Article

Not peer-reviewed version

Deep-ABSA: A Multichannel Deep Learning Framework For Aspect-Based Bangla Sentiment Analysis

[Md Morshedul Islam](#) , [G.M. SAKHAWAT HOSSAIN](#) ^{*} , [Dhiman Sarma](#) ^{*} , Rishita Chakma ^{*} , Md Mynoddin ^{*}

Posted Date: 18 July 2023

doi: 10.20944/preprints202307.1204.v1

Keywords: ABSA; word embedding; Bi-LSTM; attention; character embedding



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Deep-ABSA: A Multichannel Deep Learning Framework for Aspect-Based Bangla Sentiment Analysis

Md Morshedul Islam ¹, G.M. Sakhawat Hossain ^{2,*}, Dhiman Sarma ^{3,*}, Rishita Chakma ^{4,*} and Md Mynoddin ^{5,*}

¹ Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati, Bangladesh; riaadmorshed@gmail.com

² Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati, Bangladesh

³ Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati, Bangladesh

⁴ Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati, Bangladesh

⁵ Department of Computer Science and Engineering, Rangamati Science and Technology University, Rangamati, Bangladesh

* Correspondence: gmsakhawathossain@gmail.com (G.M.S.H.); dhiman001@gmail.com (D.S.); rishita.eee@gmail.com (R.C.); mmynoddin@gmail.com (M.M.)

Abstract: Nowadays people express their opinions on social media. Also provides product reviews on eCommerce websites and responds to various news as comments. It is necessary to know the polarity and aspect of various posts and comments for business, education and security. Aspect-based sentiment analysis (ABSA) predicts text category and polarity. In this paper, we proposed a deep learning framework Deep-ABSA for aspect-based sentiment analysis from Bangla texts. Our proposed framework is a multi-channel architecture. We implemented the word embedding, Bi-LSTM with the attention mechanism for one channel. And for another channel, we adopted the character-embedded convolutional neural network. Finally, we concatenated both channels for adjoining the features of each channel. We obtained an adequate performance from our proposed framework for aspect term analysis from Bangla sentences.

Keywords: ABSA; word embedding; Bi-LSTM; attention; character embedding

1. Introduction

Aspect-based sentiment analysis (ABSA) is extracting opinion from a user's written comments, reviews, posts or text. We can also mine opinions from audio or video. Opinion mining is very important from a business, political, or security perspective. Business quality is reflected in the comments of customers on online platforms. The popularity of a political party or a celebrity can be measured by people's opinions on social media. Sentiment analysis targets to find out the biased information from the text. However, aspect-based sentiment analysis outputs the topic name from the text as well as the polarity of a sentence. For example, "Today Bangladesh won the game and we are proud of our tigers". The aspect of this sentence is Cricket and polarity is positive.

Most of the previous works concentrated on straightforward sentiment analysis especially in Bangla language [1]. But, to obtain a fine-grained analysis aspect terms need to be extracted from a Bangla sentence to identify the sentiment more clearly. Thus, complex opinions can be discovered from the reviews of a low-resource language like Bangla [1].

Aspect-based sentiment analysis can be divided into three categories: word, sentence, and document [2]. ABSA can be done by supervised learning or unsupervised learning. For unsupervised methods, researchers use lexicon, dictionary, or corpus-based methods [3] whereas supervised

technique includes several machine learning and deep learning approaches where data labeling are performed manually. However, a few works have been carried out to extract aspect terms in the Bangla language. The main objective of this paper is to determine the aspect of term-based sentiment from Bangla texts. To attain our objective, we introduced a novel technique Deep-ABSA to perform aspect-based sentiment analysis from Bangla sentences.

The rest of the paper is arranged as follows. In Section 2, we describe the previous works related to this work. Section 3 is focused on explaining the proposed methodology whereas the result and discussions are elaborated in Section 4. Finally, we conclude this work in Section 5.

2. Related Works

Bangla language processing is now popular among researchers. Several datasets and a lot of research papers have been published on this topic. Sentiment analysis (SA) and Aspect based sentiment analysis (ABSA) also are popular topics of interest for researchers. In every language researchers proposed, to explore many techniques including machine learning, and deep learning. In this section, we summarize research works on text classification, especially on SA and ABSA. Naim and Forhad [4] proposed an equation for giving priority to every sentence in the dataset and based on this priority value they calculated TF-IDF and finally applied KNN, SVM, LR, RF, and CNN for aspect extraction. The authors in [5] used Naïve Bayes and Topical Approaches for Bangla emotion classification. They worked on a Bangla emotion classification dataset. The target is six types of emotion. They got 90% accuracy for topical approaches.

In [6], the authors trained a Bangla cricket and restaurant dataset. They used CNN for this text classification task. They got 51% and 64% f1-score for the cricket dataset and restaurant dataset respectively. Bodini and Matteo [7] implement three autoencoders for a Bangla Dataset. These are standard AEs, contractive AEs (CAEs), and sparse AEs (SAEs). They claimed the autoencoder had never been used in Bangla for aspect extraction. So these are new things for experiments in Bangla [7]. In [8] the authors found the best accuracy for Bangla Cricket and Restaurant dataset. They used pre-trained word embeddings and fed them to LSTM. The work in [9] presented a weakly supervised topic modeling method proposed for aspect-based sentiment analysis. In [10] the authors take in part-of-speech(pos) tags with word vectors. Then they use convolutional neural networks for aspect extraction. The authors in [11] applied Naïve Bayes for extracting sentiment from the Bangla book review dataset. In [12–14] used LDA topic modeling for aspect extraction. Using LSTM the work in [15] proposed a deep learning model for Bangla and Romanized Bangla Text. Srujan et al. [16] explored an RF on amazon book reviews. For TF-IDF they got 90.15% accuracy which is good. A character-level GRU is explored in the work [17].

The authors collected data from Facebook pages. They got better accuracy than the baseline word level experiment. The authors in [18] have analyzed different RNN architectures. They explored 3 datasets and got the best outcomes for GRU. In the work, [19] the authors proposed an architecture concatenating deep RNN with Bi-LSTM. Authors found their technique to be 85% accurate while extracting sentiment from manually collected facebook comments. The work in [20] proposed a model for emotions and sentiment prediction. The authors used the youtube Bangla comments dataset and applied SVM, CNN, and NB and got 65.97% accuracy for sentiment and 54.24% for emotion classification. Most of the papers in Bangla implement well-known deep learning and machine learning algorithms. Some papers in Bangla emphasize new tricks and techniques for text preprocessing. In Bangla, we noticed that people often make spelling mistakes while writing online posts or comments. So, while we use word embeddings or TF-IDF technique for word vectorization the out-of-vocabulary words or low-frequency words increase. This is because simple spelling mistakes make a word completely different for machine learning models. So, we used the sentence at the character level and also leveraged the word embedding. This gives our proposed model both local and overall words and character relationships in a sentence and ensures an adequate performance in aspect-based sentiment analysis. In the following section, the methodology used in this paper will be discussed.

3. Methodology

To classify the aspect-based sentiment from Bangla, we proposed a novel deep learning model which is presented in Figure 1. The details of our methodology are discussed in the following part.

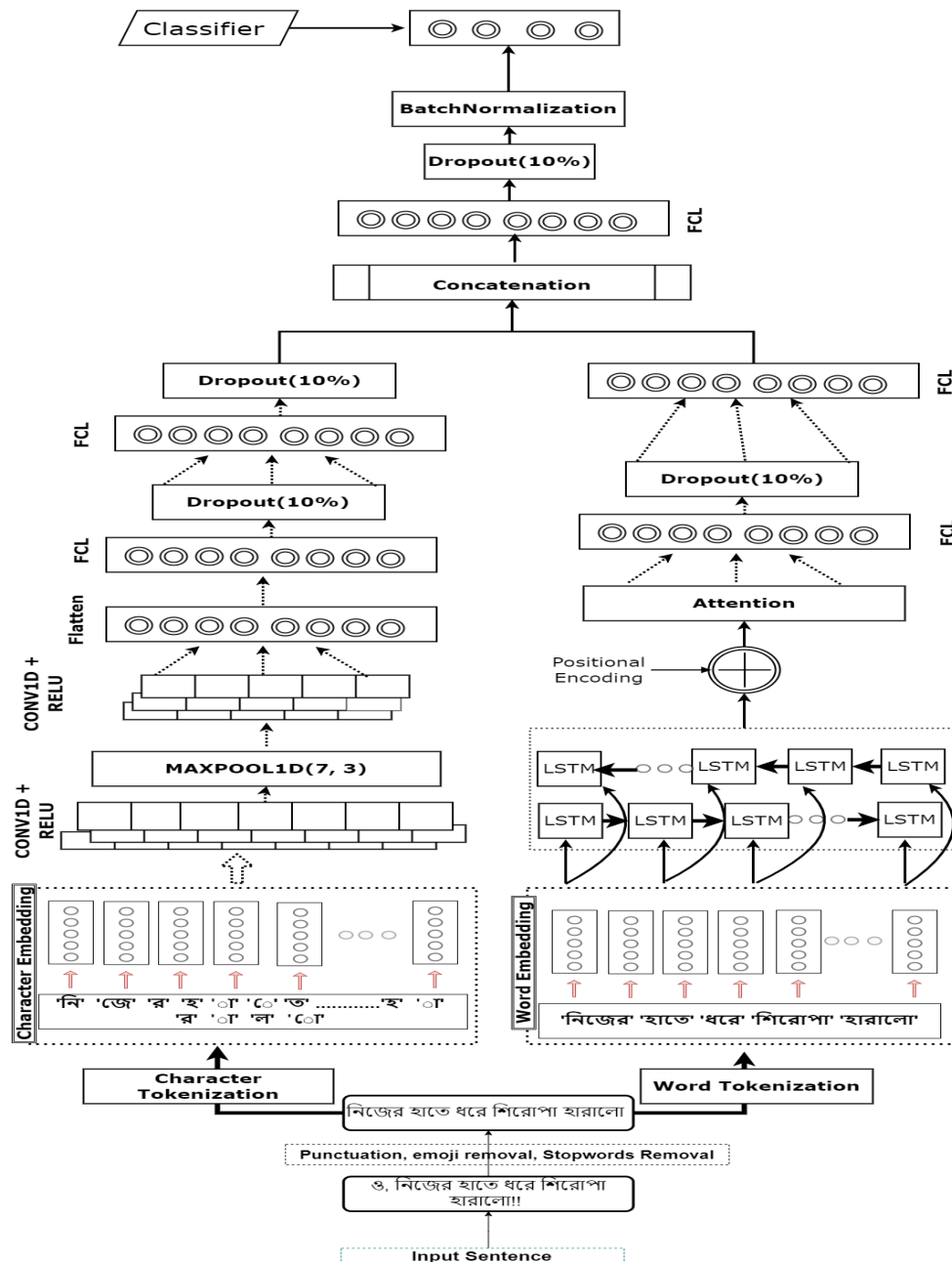


Figure 1. Proposed framework Deep-ABSA for aspect based sentiment analysis.

3.1. Proposed Model

In our model, we feed input in two channels. One channel is Word Embeddings \Rightarrow Bi-LSTM \Rightarrow Positional Encoding \Rightarrow Attention. Another channel is Character Embeddings \Rightarrow CNN \Rightarrow Max-Pooling. Word Embeddings capture the word's context and LSTM holds the contextual information for long sequences. Attention mechanism gives attention to use words and positional encoding holds the information about the order of words in sentences. Character Embeddings and CNN capture the local information. Then we concatenated both the channels for adjoining local and overall features.

3.2. Word Embedding with Bi-LSTM and Attention

In our dataset maximum number of words in a sentence is 40. We tokenized sentences. Then assign a value for every unique word and make each sentence the same length by post padding zero. In our dataset, the total sentences are 9015 and the vocabulary size is 13533.

For the word embeddings layer, we used the pre-trained word2vec model and the size of embeddings is 100. We froze the embedding layer for words. Word embeddings are numeric vector representations of words. Similar words have a similar vector representation. This similarity depends on the co-existence of words in sentences. There might be a situation where opposite words may have similar word vectors. We used the pre-trained word2vec model for generating feature vectors for every word. First, we train our dataset on the Bangla word2vec model then retrain it with the gensim word2vec model.

For every hidden layer and the output layer, RELU and SOFTMAX activation function is used respectively. Embedding layers get tokens as input and convert these tokens to their corresponding feature vector. These feature vectors are fed into the Bi-LSTM layer. Bi-LSTM processes the whole sequence forward and backward and captures the context of each input sentence. LSTM [21] is designed in a way to overcome the long-term dependency problem.

Tensors from the Bi-LSTM layer are fed to a positional encoding [22] layer and then to the attention layer. The attention layer gives attention to only necessary words whose context is important for this task. Positional encoding is a clever implementation for tracking word order in a sentence which is used as input to an attention layer. The output of the positional encoding layer is a matrix. Each row of the matrix represents a word vector with positional information. If we have an input sentence X with length L . If we need positional encoding of k_{th} words then the positional encoding is given by:

$$P(k, 2i) = \sin\left(\frac{k}{n^{2i/d}}\right) \quad (1)$$

$$P(k, 2i + 1) = \cos\left(\frac{k}{n^{2i/d}}\right) \quad (2)$$

In “(1)” and “(2)”, ‘ k ’ is the word position. The value of k is greater than or equal to zero and less than $L/2$. ‘ d ’ is the dimension of the feature vector. The value of n is defined by the user. In [22] the authors set it to 10,000. And ‘ i ’ is used for mapping indexes.

We used a scaled dot-product attention [22] mechanism. For a sentence $x [w_1, w_2, w_3, \dots, w_n]$. Scaled dot-product attention Equation (2) is:

$$\text{attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right) \mathbf{V} \quad (3)$$

In “(3)”, V, K, Q denote value, key and query respectively.

3.3. Character Embedding CNN

In our dataset, the maximum number of characters in a sentence is 160 but most of the sentences have several characters below 80. So, we took 150 as the length of the sentence and post pad smaller length sentences and truncate the larger sentences. Like words, we tokenized every character. The total number of unique characters was 81. So, Vocabulary size is much smaller than word embedding. Character Embedding is generated at the character level. For example, the word “Bangla” is split into ‘B’, ‘a’, ‘n’, ‘g’, ‘l’, and ‘a’. A Sequence of character tokens is fed into the embedding layer and initially, the feature vector is generated randomly. Character embedding doesn’t capture the deep meaning but gathers lexical information and local relationship. The output vector from the embedding layer is fed into the 1D convolutional layer. Character-based CNN [23] received the character feature vectors as input. Each convolution layer generates different features by applying different filters on character feature vectors. Three Conv1D layers of 256 neurons are used in our architecture. The filter size is

7 and the sliding window is 3 in every convolution. After every Conv1D we used a Maxpool layer where the pool size is 3. Then flatten the output of the CNN layer and add two dense layers of size 256 neurons and 10% dropout between dense layers. Finally, we concatenated both channels. After concatenating this model captured the relational information from the LSTM-Attention channel and local relationship from the Character CNN channel. The concatenated feature is fed into a dense layer with a dropout of 10% and Batch Normalization [24] layer. In the output layer, 4 neurons have been used for aspect and 2 neurons for sentiment classification. In the following section, we discuss the result analysis obtained in this paper.

4. Result Analysis

4.1. Dataset Description

In this research, we used a publicly available dataset BAN-ABSA [25]. Table 1 describes the summary of our experimental dataset. This dataset contains unnecessary characters (punctuation marks, emojis, digits, tags, and URL) which do not have any impact on aspect or polarity. We removed these unnecessary characters. Cleaned data makes the machine learning models more efficient and reduces the error rate. Stopwords are common words in a language and do not have any contribution to describing the aspect. We also remove the stop words using BNLP (Bangla NLP Library) stop words list. Some stop words directly indicate a sentence's polarity. We white-listed these stop words. We also removed some most frequent words which have no impact on aspect or polarity. People use both English and Bangla characters in the same sentence. We removed these English words. Because our concern for this research is the Bangla language.

4.2. Model Compilation

We compiled our model using the 'categorical_crossentropy' loss function and adam optimizer and the learning rate of the model is 0.001. We took batch size 64. For the hidden layer, we used RELU, and for the output/prediction layer SOFTMAX. We ran every model with 100 epochs. We have splitted the dataset into train and test by 80% and 20%.

Table 1. Summary of our experimental dataset.

Overall descriptions of our experimental dataset	
Dataset Name	BAN-ABSA[25]
Aspects	other, sports, political and religion
Polarity	Positive, Negative and Neutral
Total Sentences	9015
Total Words	63,665
Max Char in a sentence	160
Average Char in a sentence	33
Max words in a sentence	40
Average words in sentence	5

4.3. Performance measurement

We used four performance metrics to measure and compare our proposed model with state-of-the-art models. The performance metrics and their equation is given below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \quad (7)$$

4.4. Aspect Term Result Analysis

Table 2 shows the performance comparison of our proposed model with other popular well-known techniques. We implemented the channels of our proposed model separately to clarify how well they perform to identify aspect terms from Bangla sentences. Finally, we concatenated both channels and executed them to obtain improved performance for aspect-based sentiment analysis from Bangla sentences. The first channel Word Embeddings \Rightarrow Bi-LSTM \Rightarrow Positional Encoding \Rightarrow Attention provided the 77% accuracy with 78% f1-score for aspect term analysis. On the other hand, the second channel Character Embeddings \Rightarrow CNN \Rightarrow Max-Pooling pulled up about 81% accuracy with 81% f1-score. The second channel performed better than the first one. This is perhaps because of the use of the character embedding technique that we introduced in our proposed model. Secondly, the use of a one-dimensional convolutional layer helped to develop more accurate feature vectors. After concatenating both the channels we obtained an average accuracy of 81% with slightly better precision and f1-score of 84% and 82% respectively from our proposed Deep-ABSA framework for aspect-based sentiment analysis.

To justify the performance of our proposed model, we also implemented several machine learning and deep learning approaches with a variety of combinations. Table 2 expresses the comparison of their results of them. Among machine learning approaches, Logistic Regression was found to be better alongside TF-IDF technique for aspect-based Bangla sentiment analysis.

Table 2. Comparison table: state-of-the-art vs proposed for aspect term.

Embedding vector	Model Name	Precision	Recall	F1 Score	Acc
word2vec	LSTM	.82	.80	.81	.80
word2vec	GRU	.81	.80	.80	.79
word2vec	CNN+Bi-LSTM	.78	.78	.78	.77
word2vec	CNN+Bi-GRU	.77	.77	.77	.76
Character Embedding	Character CNN	.82	.81	.81	.81
Character Embedding	Character CNN + Bi-LSTM	.81	.81	.81	.81
Character Embedding	Character CNN + Bi-GRU	.80	.80	.80	.80
word2vec	Positional Encoding + Attention	.77	.76	.77	.76
word2vec	Bi-LSTM + Attention	.78	.78	.78	.77
tf-idf	SVM	.85	.71	.75	.74
tf-idf	MNB	.82	.75	.78	.76
tf-idf	LR	.85	.75	.78	.77
word2vec, char Embd	Proposed Model	.84	.83	.82	.81

Figure 2 depicts the training and validation accuracy vs epoch curve of our proposed model for aspect-based sentiment analysis. From the curve, it is clear that our proposed model achieved adequate performance during the training and validation session. Figures 3 and 4 represent the ROC curve and precision-recall curve of our proposed model on our experimental dataset. Both of the curves express how well our proposed model performed to extract aspect-based sentiment from the dataset.

Figure 5 is the confusion matrix of our proposed model on aspect-based analysis. This matrix shows that our model is slightly behind in identifying the “other” category. This is because many general sentences are even confusing for humans to categorize. But our model worked well for categorizing the rest of the aspect terms from the dataset. Finally, we compared our proposed

Deep-ABSA model with the state-of-the-art technique implemented on the same dataset as ours and found that our model is slightly ahead in terms of performance. Table 3 shows the performance comparison between our proposed model and other state-of-the-art technique for aspect-based Bangla sentiment analysis.

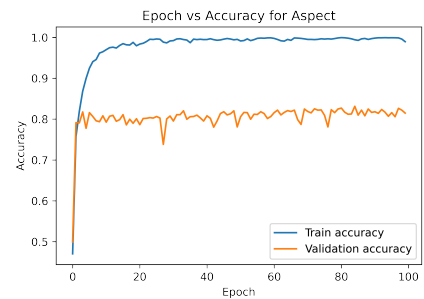


Figure 2. Training And Validation accuracy curve of Deep-ABSA model.

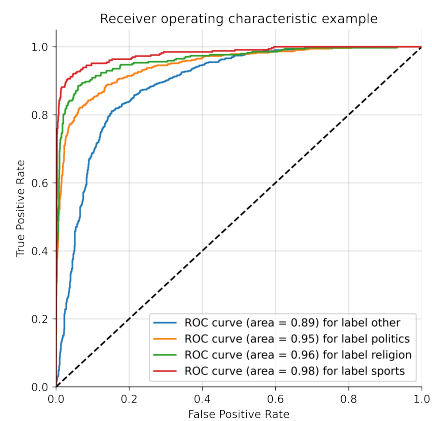


Figure 3. ROC curve of Deep-ABSA model.

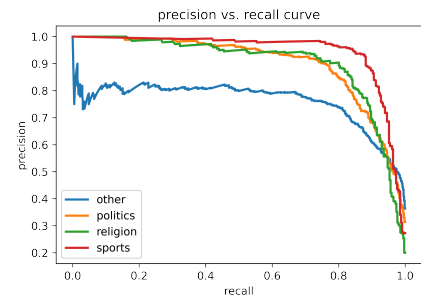


Figure 4. Precision Recall curve of Deep-ABSA model.

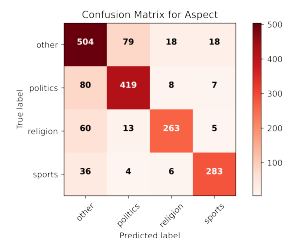


Figure 5. Confusion matrix of Deep-ABSA model.

Table 3. Comparison of our model with the state-of-the art technique for ABSA in Bangla.

Paper Name	Acc	F1 score	Pre-cision	Re-call
BAN-ABSA: An Aspect-Based Sentiment Analysis dataset for Bengali and it's baseline evaluation [25]	79	79	80	79
This paper	81	82	84	83

5. Conclusions

We developed and proposed a novel deep learning architecture Deep-ABSA for aspect-based Bangla sentiment analysis. Our proposed multichannel model captures the necessary information for word and character tokens effectively. We have shown that our Deep-ABSA model achieved a satisfactory performance compared with the other traditional machine learning and deep learning approaches as well as with the state-of-the-art technique by acquiring 81% accuracy for aspect-based sentiment analysis from Bangla sentences. However, exploring the model with the other well-known Bangla dataset can be the future scope of this study. Adopting the transformer architecture with our proposed model can be another future scope of this work.

References

1. M. A. Rahman and E. K. Dey, "Datasets for aspect-based sentiment analysis in bangla dataset". MDPI Journals, 2018, doi: 10.3390/data3020015.
2. Wang, Jie, Bingxin Xu, and Yujie Zu. "Deep learning for aspect-based sentiment analysis." 2021 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE). IEEE, 2021.
3. Abdulla, Nawaf A., et al. "Arabic sentiment analysis: Lexicon-based and corpus-based." 2013 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT). IEEE, 2013.
4. Naim, Forhad An. "Bangla aspect-based sentiment analysis based on corresponding term extraction." 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD). IEEE, 2021.
5. Tuhin, Rashedul Amin, et al. "An automated system of sentiment analysis from Bangla text using supervised learning techniques." 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS). IEEE, 2019.
6. Rahman, Md Atikur, and Emon Kumar Dey. "Aspect extraction from Bangla reviews using convolutional neural network." 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR). IEEE, 2018.
7. Bodini, Matteo. "Aspect Extraction from Bangla Reviews Through Stacked Auto-Encoders." Data 4.3 (2019): 121.
8. Wahid, Md Ferdous, Md Jahid Hasan, and Md Shahin Alom. "Cricket sentiment analysis from Bangla text using recurrent neural network with long short term memory model." 2019 International Conference on Bangla Speech and Language Processing (ICBSLP). IEEE, 2019.
9. Lu, Bin, et al. "Multi-aspect sentiment analysis with topic models." 2011 IEEE 11th international conference on data mining workshops. IEEE, 2011.
10. Poria, Soujanya, Erik Cambria, and Alexander Gelbukh. "Aspect extraction for opinion mining with a deep convolutional neural network." Knowledge-Based Systems 108 (2016): 42-49.
11. Hossain, Eftekar, Omar Sharif, and Mohammed Moshui Hoque. "Sentiment polarity detection on bengali book reviews using multinomial naive bayes." Progress in Advanced Computing and Intelligent Engineering. Springer, Singapore, 2021. 281-292.
12. Moghaddam, Samaneh, and Martin Ester. "On the design of LDA models for aspect-based opinion mining." Proceedings of the 21st ACM international conference on Information and knowledge management. 2012.
13. Kiritchenko, Svetlana, et al. "Nrc-canada-2014: Detecting aspects and sentiment in customer reviews." Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014). 2014.

14. Brychcín, Tomáš, Michal Konkol, and Josef Steinberger. "Uwb: Machine learning approach to aspect-based sentiment analysis." *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. 2014.
15. Hassan, Asif, et al. "Sentiment analysis on bangla and romanized bangla text using deep recurrent models." *2016 International Workshop on Computational Intelligence (IWCI)*. IEEE, 2016.
16. Srujan, K. S., et al. "Classification of amazon book reviews based on sentiment analysis." *Information Systems Design and Intelligent Applications*. Springer, Singapore, 2018. 401-411.
17. Haydar, Mohammad Salman, Mustakim Al Helal, and Syed Akhter Hossain. "Sentiment extraction from bangla text: A character level supervised recurrent neural network approach." *2018 international conference on computer, communication, chemical, material and electronic engineering (IC4ME2)*. IEEE, 2018.
18. Baktha, Kiran, and B. K. Tripathy. "Investigation of recurrent neural networks in the field of sentiment analysis." *2017 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2017.
19. Sharfuddin, Abdullah Aziz, Md Nafis Tihami, and Md Saiful Islam. "A deep recurrent neural network with bilstm model for sentiment classification." *2018 International conference on Bangla speech and language processing (ICBSLP)*. IEEE, 2018.
20. Tripto, Nafis Irtiza, and Mohammed Eunus Ali. "Detecting multilabel sentiment and emotions from bangla youtube comments." *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*. IEEE, 2018.
21. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." *Neural computation* 9.8 (1997): 1735-1780.
22. Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).
23. Zhang, Xiang, Junbo Zhao, and Yann LeCun. "Character-level convolutional networks for text classification." *Advances in neural information processing systems* 28 (2015).
24. Ioffe, Sergey, and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." *International conference on machine learning*. PMLR, 2015.
25. Masum, Mahfuz Ahmed, et al. "BAN-ABSA: An Aspect-Based Sentiment Analysis dataset for Bengali and it's baseline evaluation." *arXiv preprint arXiv:2012.00288* (2020).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.