Int. J. Nonlinear Anal. Appl.
Volume 12, Special Issue, Winter and Spring 2021, 29-38
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/IJNAA.2021.4769

Ensemble Deep Learning for Aspect-based Sentiment Analysis

Azadeh Mohammadi[a*], Anis Shaverizade [b]

[a] *Assistant Professor, Computer Department, University of Isfahan, Isfahan, Iran.*
[b] *Graduate student, IT and Computer Department, Sepahan Institute of Higher Education, Isfahan, Iran.*

(Communicated by Dr. Ehsan Kozegar)

Abstract

Sentiment analysis is a subfield of Natural Language Processing (NLP) which tries to process a text to extract opinions or attitudes towards topics or entities. Recently, the use of deep learning methods for sentiment analysis has received noticeable attention from researchers. Generally, different deep learning methods have shown superb performance in sentiment analysis problem. However, deep learning models are different in nature and have different strengths and limitations. For example, convolutional neural networks are useful for extracting local structures from data, while recurrent models are able to learn order dependence in sequential data. In order to combine the advantages of different deep models, in this paper we have proposed a novel approach for aspect-based sentiment analysis which utilizes deep ensemble learning. In the proposed method, we first build four deep learning models, namely CNN, LSTM, BiLSTM and GRU. Then the outputs of these models are combined using stacking ensemble approach where we have used logistic regression as meta-learner. The results of applying the proposed method on the real datasets show that our method has increased the accuracy of aspect-based prediction by 5% to 20% compared to the basic deep learning methods.

Keywords: Deep Learning, Ensemble Learning, Natural Language Processing, Opinion Mining, Sentiment Analysis

1. Introduction

One of the needs that human beings always feel is the need to understand the behavior, opinions and beliefs of other people. Recent advances in web technologies have provided new ways of communication, including social networks, blogs, e-commerce websites, and more. According to the considerable

*Corresponding Author: Azadeh Mohammadi
Email address: az.mohammadi@eng.ui.ac.ir (Azadeh Mohammadi[a*], Anis Shaverizade [b])

Received: October 2020 *accepted:* December 2020

amount of data obtained from these platforms, the need for an automated system to organize and analyze this volume of data is ever-increasing.

One of the main processes on this type of data is sentiment analysis (opinion mining), which aims to extract the user's attitude and feelings from the comments or text written by him [11]. Sentiment analysis has different applications. For example, by extracting opinion from comments, business owners can get important information from customer feedbacks and consequently improve the product quality and customer service [12]. In addition, sentiment analysis can be used in other areas including analysis of political issues and film reviewing [5].

Sentiment analysis can be performed at three levels: document level, sentence level and aspect level. At the document level, the whole text is considered as an information unit or a subject, and the polarity of the text (positive, negative, or neutral) is analyzed and extracted for the whole text [13]. In sentence level sentiment analysis, the emotion expressed in each sentence will be classified [3].

The problem with document/ sentence based methods is that in these methods it is assumed that only one subject is expressed in the document/sentence, but in many cases this is not true. For example, in the sentence "My mobile phone has a high quality screen, but the battery life is very short.", simultaneously, two positive and negative opinions are expressed. If we consider "screen" aspect the polarity is positive but the opinion about "battery" aspect is negative. Therefore, for a more accurate analysis, we should consider the entities and their related aspects in a text and classify the polarity at aspect level. This is called aspect-based sentiment analysis [9].

Alvarez-Lopez et al. [1] proposed a CRF and SVM-based model for aspect-based sentiment classification, but the method failed to accurately extract the polarity of opinions. Over the past decade, deep learning methods have achieved many successes in various fields, including Natural Language Processing (NLP). In this regard, many researchers have used deep learning methods for the sentiment analysis problem recently [22, 25].

Xu et al. [21] utilized a Convolutional Neural Network (CNN) model for aspect-based sentiment classification. Wang et al. [18] proposed an Attention-based Long Short-Term Memory (LSTM) model for aspect-based sentiment analysis. The attention mechanism can concentrate on different parts of a sentence when different aspects are taken as input. Xing and Xiao in [19] have used an attention based Gated Recurrent Unit (GRU) model for aspect based sentiment classification. Clematide and Simon in 2018 [8] presented a bidirectional LSTM (BiLSTM) architecture with a multilayer perceptron on top that dynamically mixes word and character-level representations. Zhou et al. [27] used two attention based LSTM network for cross-language sentiment classification to model the word sequences in the source language (Chinese) and target language (English).

The problem with convolutional model is that these model do not take into account the order of the words in the sentence, so it cannot extract proper meanings from sentences. This issue can be addressed with recurrent (memory-based) models, such as LSTM or GRU, which can learn long term and syntactic dependencies. On the other hand, memory-based models cannot perform accurate analysis if the order of words in a sentence changes [23].

According to the strengths and limitations of each deep learning classifier, in this paper we proposed an ensemble learning method which use deep learning models as base classifier and combine the outputs of these models with the stacking ensemble approach. In fact, in the proposed method, first we create and train four deep learning models, namely CNN, LSTM, BiLSTM and GRU. Then a logistic regression model is used as a meta-learner to combine the outputs of base classifiers. To the best of our knowledge this is the first time that a meta-learner is utilized to integrate different deep learning models in aspect-based sentiment analysis problem.

In the rest of the paper, we first provide some necessary background in Section 2. Then, in Section

3, the proposed method is described and the experimental results are demonstrated and discussed in Section 4. Finally, the conclusion and future works are explained in section 5.

2. Background

In this section we briefly describe deep neural network models and ensemble learning.

2.1. Convolutional Neural Network model

Convolutional Neural Network (CNN) is a specialized type of deep learning model which is inspired by the organization of animal visual cortex [26]. CNN is composed of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN consist of a series of convolutional layers which play a key role in CNN. The convolution layer is composed of a set of independent filters. Each filter is independently convolved with the input and the result is usually pooled or subsampled to smaller dimensions and fed to the next layer. The convolution layers extract local feature from input and finally a fully connected layer make the final prediction.

2.2. Recurrent Neural Network model

Recurrent Neural Network (RNN) is a type of deep learning model which introduces recurrent connections. The recurrent connections allow the network to retain previous outputs as memory in the internal states. Consequently, RNN can store information about what has been processed so far and can be employed for processing sequential data [6].

2.3. Long Short-Term Memory model

Since RNN is trained by backpropagation through time, the gradient signal gets smaller and smaller as it backpropagates further. For long sequences the gradient approaches to zero and the weights will not be updated. This problem is called vanishing gradient problem [16].

Long Short-Term Memory (LSTM) is a special kind of RNN that alleviates the problem of vanishing gradients. It is composed of a memory cell and three gates namely input gate, output gate and forget gate which regulate the flow of information in the network. If the forget gate is on and the input and output gates are off, memory cell just passes the gradients without change. Therefore, LSTM reduces the vanishing gradient problem and can learn long-term dependencies [16].

2.4. Bidirectional LSTM model

Bidirectional LSTM (BiLSTM) model is a kind of recurrent neural networks which consists of two LSTMs: one taking the input in a forward direction, and the other in a backwards direction. Consequently, it gets more information from text by knowing what text follow and precede a word in a sentence, by processing the sequence in two directions [20].

2.5. Gated Recurrent Unit model

Gated Recurrent Unit (GRU) is similar to LSTM model but it has fewer parameters than LSTM. Instead of three gates in LSTM, GRU has two gates namely update gate and reset gate. The update gate controls information that flows into memory, and the reset gate controls the information that flows out of memory. If the update gate is 1, the previous memory is fully preserved, and if it is 0, the previous memory is completely forgotten [7].

As mentioned before, GRU has less parameters and hence it is faster than LSTM but as it is demonstrated in [24] their performance depends on the characteristics of the datasets.

2.6. Ensemble learning

In recent years, ensemble learning has been considered as one of the most successful techniques in machine learning [14]. Ensemble learning combines several machine learning techniques into one predictive model in order to increase the performance in comparison to single models (base classifiers). Most ensemble methods use a single base learning algorithm to produce homogeneous base learners, i.e. learners of the same type. There are also some methods that use heterogeneous learners, i.e. learners of different types [14].

Combining base classifiers can be done in different ways, including voting [2], averaging [4], bagging [14], boosting [10] and stacking [15] which we will explain in the following.

In the voting approach, every individual classifier votes for a class and the final prediction is made by considering all votes. In the majority voting, the final prediction is equal to the output which is chosen by the most classifiers. A limitation of the majority voting is that it treats all models the same, i.e. all models contribute equally to the prediction. To address this problem, we can use weighted voting, where the importance of models in voting is different [2]. The voting approach usually reduces overfitting and creates a smooth model but it is not guaranteed to provide better performance than any single model used in the ensemble [2].

In the averaging method, the final prediction is equal to the average output of the base classifiers. Similar to voting, averaging can be done in weighted version. The averaging method is used when the outputs are continuous [4].

Bagging is a homogenous ensemble method where same base learners are trained in parallel on different random subsets of the training set. To obtain the data subsets bootstrap sampling is used. Bagging method reduces variance and improves stability where the base classifiers are unstable [13].

Boosting is a homogenous ensemble method like bagging, but in contrast to bagging it is a sequential process where each subsequent model attempts to correct the errors of its predecessor. In order to improve the accuracy of models, more weight is given to samples that were misclassified by earlier rounds. Boosting method decreases the bias error and produces strong predictive model [10].

Stacking is a heterogeneous ensemble learning technique that combines multiple classification models via a meta-classifier. The base classifiers are trained based on a complete training set, then the meta-model is trained on the outputs of the base level model as features [15]. The superiority of stacking is that it can combine the advantages of distinct machine learning algorithms on a classification task and make predictions that have better performance than any single model in the ensemble.

3. Methods

In this paper, we proposed a novel method for aspect level sentiment analysis problem which is based on stacked ensemble learning. The proposed method utilizes four deep learning model namely CNN, LSTM, BiLSTM and GRU as base classifiers and then combines the outputs using a meta classifier. Combining different deep learning models allows us to exploit the structural and functional advantages of each model and improve the total performance. In the following we explain the base learners and the meta classifier in more details.

The first base classifier we used in this paper is a LSTM network. The LSTM model has a good ability to keep the sequential information of sentences and is very powerful for modeling long texts and extracting their meanings [17]. The LSTM model in our paper composed of three layers namely embedding, LSTM and dense layer.

The embedding layer maps the input words to vector of numbers such that words which have a similar meaning in the context are embedded next to each other. Before training we removed punctuations and rare words. The size of our vocabulary is 10000. We can fold each word in just as many dimensions as we want. We considered the size of embedding vector equal to 32. The next layer of our LSTM model is the LSTM layer which is composed of 128 neurons. This layer has the ability of capturing the sequential data by considering the previous data.

The last layer of our LSTM classifier is a dense layer with 3 neurons which determines the sentiments of comments as positive, negative or neutral. The activation function in the last layer is softmax. The softmax function returns a number between 0 to 1 for each class which show the probability of target classes. We applied Adam optimization method for updating weights in the network and used categorical-crossentropy as loss function.

The second base classifier we used is a GRU network. GRU has an update gate which determines how much of the past information (stored in the previous hidden state) needs to be retained for the future and a rest gate which determines how to combine the new input with the previous memory. Our GRU network is composed of three layers namely embedding, GRU and dense layer. The embedding layer is similar to our LSTM model. The GRU layer is composed of 128 neurons. We used tanh as activation function. The loss function and optimization methods are categorical-crossentropy and Adam method, respectively.

In addition to LSTM and GRU, we used a BiLSTM model as our third base classifier. Since BiLSTM model traverse the text in two directions, we not only consider previous words in a sentence but also the next words for extracting meanings. Like the previous two models, our BiLSTM network has 3 layers namely embedding, BiLSTM layer and dense layer. Embedding and dense layers are similar to our LSTM network. The BiLSTM layer consists of 128 neurons with tanh activation function. For training the model we use categorical-cross entropy as loss function and Adam as optimization method.

Our last base classifier is a CNN network. CNN can extract key features from the text automatically. The first layer of our CNN network is an embedding layer which is responsible for vectorization of words. The next layer is a convolutional layer. The results of convolution of inputs and kernels are given to a ReLU function. After that a GlobalMaxPooling1D is used for dimension reduction. Then we have two dense layers with 16 and 3 neurons, respectively. We use categorical-crossentropy as loss function and Adam method as optimization method.

After generating the base classifiers, we use an ensemble learning method to exploit the advantages of different models which we mentioned earlier. In the following, we describe the structure of our models more precisely.

Our proposed model is based on stacking ensemble approach where we train our base classifiers, i.e. LSTM, GRU, BiLSTM and CNN; then the outputs of these models are combined using stacking method. In this paper we use multinomial logistic regression for combining the output of the above-mentioned classifiers. The outline of the proposed model is displayed in figure 1. In this figure, x shows the sample and O1 to O4 shows the output of base classifiers. y is the final output of our model which is obtained from the combination of O1 to O4.

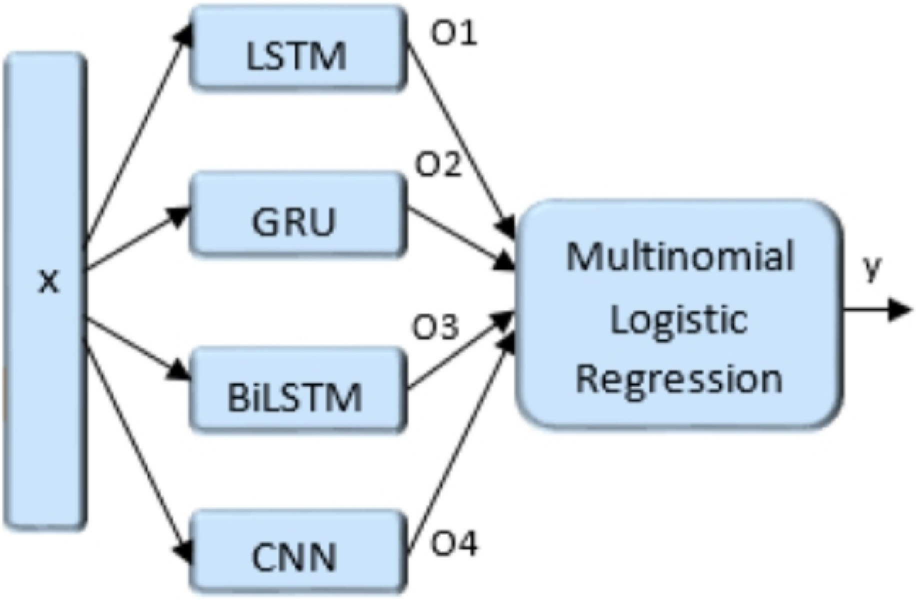

Figure 1: The outline of our proposed model

Since the multinomial logistic regression get numeric input and the outputs of our base classifiers are in the form of one-hot vector, first we should convert the output of each classifier to a number. In this regard, we find the maximum value of each output vector. Then we return the index of the maximum value for each classifier as a number, which can be 0, 1 or 2. In this coding, 0 is corresponding to neutral polarity and 1 and 2 shows negative and positive polarity, respectively. The obtained numbers are then given as input to the multinomial logistic regression. Each of these numbers are linked to multinomial logistic regression model with a different weight which indicates the importance of each base classifier in producing the final output (y). These weights are adjusted during training.

In fact, [O1, O2, O3, O4], which we call it x_{lr}, is injected into a logistic regression model as input. Since in our paper, sentiments are classified in three classes namely neutral, negative and positive, the final output, y, is a 3-dimensional vector which determines the polarity of sentiments. Under a multinomial logistic regression model, the probability that input x_{lr} belongs to class i is written as (1):

$$p(y_i = 1|x_{lr}, w) = \frac{exp(x_{lr}w_i)}{\sum_{j=1}^{3} exp(x_{lr}w_j)} \quad (1)$$

where w is the weight matrix and the row w_j in this matrix determines the importance of j^{th} base classifier in producing the final output. The weight matrix, w, is updated during training.

Because of the normalization condition shown in (2), we only have to estimate the probability for two classes and the probability of the third class can be computed using (2):

$$\sum_{i=1}^{3} p(y_i = 1|x_{lr}, w) = 1 \quad (2)$$

Finally, multinomial logistic regression assigns each sample to a class where it has the greatest possibility of belonging. Then, the predicted polarity for each sample x is compared to its assigned polarity in the training dataset to determine the amount of loss. The weight matrix is updated accordingly to reduce the loss value.

4. Results and Discussion

In this section, we describe the datasets used in this work and discuss the experimental results.

4.1. Dataset

In order to evaluate the performance of our proposed approach, the model is applied on real datasets and its performance is compared with base classifiers.

We evaluated our model on two different domains, namely laptops and restaurants. These datasets are available as SemEval2016 (SE-ABSA 2016) task 5 [28].

ABSA was introduced for the first time in the context of SemEval in 2014 [29]. The SemEval 2016 Task 5 provided datasets of English reviews annotated with aspect terms and their polarity for the laptop and restaurant domains. It gives the opportunity to do aspect based sentiment analysis on reviews.

The Restaurant's dataset consists of 350 reviews (2000 sentences) for training and 90 reviews (676 sentences) for testing. The Laptop's dataset consists of 450 reviews (2500 sentences) for training and 80 reviews (808 sentences) for testing. Table 1 summarizes the characteristics of the datasets used for the evaluation of the proposed model.

Table 1: The properties of datasets

	Train		**Test**	
	#Sent	**#Reviews**	**#Sent**	**#Reviews**
Restaurant	350	2000	90	676
Laptop	450	2500	80	808

In each sentence of reviews, entities are extracted and a pair of Entity-Attribute is determined which shows the aspect category. The possible entity types and attribute labels for restaurant and laptop datasets are given in table 2 and table 3, respectively. Each pair of Entity-Aspect (aspect category) is assigned to a sentiment polarity (positive, negative or neutral). The aim is to predict the sentiment polarity in the test datasets.

Table 2: Entity and attribute labels in restaurant domain

Entity labels
RESTAURANT, FOOD, DRINKS, AMBIENCE, SERVICE, LOCATION
Attribute labels
GENERAL, PRICES, QUALITY, STYLE_OPTIONS, MISCELLANEOUS

Table 3: Entity and attribute labels in laptop domain

Entity labels
LAPTOP, DISPLAY, KEYBOARD, MOUSE, MOTHERBOARD, CPU, FANS_COOLING, PORTS, MEMORY, POWER_SUPPLY, OPTICAL_DRIVES, BATTERY, GRAPHICS, HARD_DISK, MULTIMEDIA_DEVICES, HARDWARE, SOFTWARE, OS, WARRANTY, SHIPPING, SUPPORT, COMPANY
Attribute labels
GENERAL,PRICE, QUALITY, DESIGN_FEATURES, OPERATION_PERFORMANCE, USABILITY, PORTABILITY, CONNECTIVITY, MISCELLANEOUS

4.2. Experimental results

For evaluating our proposed method, we applied it on restaurant and laptop datasets and compared the results with base classifiers (LSTM, GRU, BiLSTM and CNN). We used 30 epochs for training and considered the batch-size equal to 32. To prevent overfitting, we used Early stopping technique in all models. Other model-related parameters are specified in Section 3.

For evaluation we used precision as performance measure which is defined as (3)

$$precision = \frac{TP}{TP + FP} \quad (3)$$

In (3) TP is the True Positive and FP is the False Positive.

Since our problem is a multiclass categorization, we used macro-averaged precision which is the average of per-class precision. The precision for each class (per-class precision) is computed by (3).

The obtained results for the restaurant and laptop domains are represented in table 4 and table 5, respectively. As the results show, our proposed model which utilizes ensemble learning method outperforms the base classifiers. Among individual classifiers, GRU has shown the worst performance in restaurant as well as laptop dataset. Our proposed method has increased the precision by 10% compared to GRU in the laptop dataset and by 20% in the restaurant dataset. The best base classifier in both domains is CNN which has achieved an accuracy of 66% and 64.3% on restaurant and laptop datasets, respectively. Our proposed model improved the CNN' precision about 5% in both domains.

According to the results, combining the base deep learning methods in a stacked approach has increased the precision compared to each individual classifier. It indicates that by combining models with different functional and structural characteristics, we can utilize their advantages. In fact, using memory-based model such as LSTM, BiLSTM and GRU let the model to learn long term dependencies. LSTM, BiLSTM and GRU have different structures and they can strengthen each other when they are combined. On the other hand, memory-based models cannot perform accurately if the order of words in a sentence changes. CNN model can cover this limitation and consequently the overall performance of our ensemble method is increased.

Table 4: Performance comparison of our proposed model with base deep learning models in Restaurant dataset

Model	Precision
LSTM	60.2 %
GRU	56.8 %
BiLSTM	64.8 %
CNN	66 %
Proposed Model	69.3 %

Table 5: Performance comparison of our proposed model with base deep learning models in Laptop dataset

Model	Precision
LSTM	63.2 %
GRU	61 %
BiLSTM	63 %
CNN	64.3 %
Proposed Model	67.5 %

5. Conclusion and future work

Aspect-based sentiment analysis (ABSA) of reviews has recently gained the researchers interest. In this paper we proposed a new model for aspect-based sentiment analysis. The proposed framework is a deep neural network architecture using ensemble learning. In the proposed method, first we create and train four deep learning models, namely CNN, LSTM, BiLSTM and GRU. Then a logistic regression model is used as a meta-learner to combine the outputs of base classifiers.

We applied our method on real datasets. Evaluation results show that the proposed approach outperforms the baseline method up to 20% in terms of precision. Since the result of ensemble method is promising, for future work we plan to apply ensemble learning on other text processing task.

References

[1] T. Álvarez-López, J. Juncal-Martínez, M. Fernández-Gavilanes, E. Costa-Montenegro and F. Javier, GTI at SemEval2016- Task 5: SVM and CRF for Aspect Detection and Unsupervised Aspect-Based Sentiment Analysis, in: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, California, (2016) pp. 306–311.

[2] S. Ardabili and A. Musavi, Advances in Machine Learning Modeling Reviewing Hybrid and Ensemble Methods, Lecture Notes in Networks and Systems. 101 (2019) 215–227.

[3] R. Arulmurugan, K.R. Sabarmathi and H. Anandakumar, Classification of sentence level sentiment analysis using cloud machine learning techniques, Cluster Computing. 22 (2019) 1199–1209.

[4] M. Awni, M.I. Khalil and H.M. Abbas, Deep-Learning Ensemble for Offline Arabic Handwritten Words Recognition, in: 2019 14th International Conference on Computer Engineering and Systems (ICCES, Cairo, Egypt,(2019) pp. 40–45.

[5] E. Camberia, Affective computing and sentiment analysis, IEEE Intelligent Systems. 31 (2016) 102–107.

[6] E. Can, A. Ezen-can and F. Can, Multilingual Sentiment Analysis: An RNN-Based Framework for Limited Data, in: ACM SIGIR 2018 Workshop on Learning from Limited or Noisy Data, (2018).

[7] Y. Cheng, L. Yao, G. Xiang, G. Zhang, T. Tang and L. Zhong, Text Sentiment Orientation Analysis Based on Multi-Channel CNN and Bidirectional GRU With Attention Mechanism, IEEE Access. 8 (2020) 134964–134975.

[8] S. Clematide, A Simple and Effective biLSTM Approach to Aspect-Based Sentiment Analysis in Social Media Customer Feedback, in: 14th Conference on Natural Language Processing, Vienna, Austria, 2018: pp. 29–33.

[9] H.H. Do, P. Prasad, A. Maag and A. Alsadoon, Deep Learning for Aspect-Based Sentiment Analysis: A Comparative Review, Expert Systems With Applications. 118 (2019) 272–299.

[10] Y. Freund, RESchapire, A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting, Journal of Computer and System Sciences. 55 (1997) 119–139.

[11] D. Gamal, M. Alfonse, E.-S. M. El-Horbaty and A.-B. M. Salem, Analysis of Machine Learning Algorithms for Opinion Mining in Different Domains, Machine Learning and Knowledge Extraction. 1 (2019) 224–234.

[12] P. Liu, S. Joty and H. Meng, Fine-grained Opinion Mining with Recurrent Neural Networks and Word Embeddings, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, (2015) pp. 1433–1443.

[13] G. Rao, W. Huang, Z. Feng and Q. Cong, LSTM with sentence representations for document-level sentiment classification, Neurocomputing. 308 (2018) 49–57.

[14] O. Sagi and L. Rokach, Ensemble learning: A survey, Data Mining and Knowledge Discovery. 8 (2018) 1–18.

[15] K. Sarkar, A Stacked Ensemble Approach to Bengali Sentiment Analysis, Lecture Notes in Computer Science. 11886 (2020) 102–111.

[16] A. Sherstinsky, Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network, Physica D: Nonlinear Phenomena. 404 (2020) 1–43.

[17] K. Smagulova and A.P. James, A survey on LSTM memristive neural network architectures and applications, The European Physical Journal Special Topics. 228 (2019) 2313–2324.

[18] Y. Wang, M. Huang, L. Zhao and X. Zhu, Attention-based LSTM for Aspect-level Sentiment Classification, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, (2016) pp. 606–615.

[19] Y. Xing and C. Xiao, A GRU Model for Aspect Level Sentiment Analysis, Journal of Physics: Conference Series. 1302 (2019) 1–7.

[20] G. Xu, Y. Meng, X. Qiu, Z. Yu and X. Wu, Sentiment Analysis of Comment Texts Based on BiLSTM, IEEE Access. 7 (2019) 51522–51532.

[21] H. Xu, B. Liu, L. Shu and P.S. Yu, Double Embeddings and CNN-based Sequence Labeling for Aspect Extraction, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, (2018) pp. 592–598.

[22] A. Yadav and D.K. Vishwakarma, Sentiment analysis using deep learning architectures: a review, Artificial Intelligence Review. 53 (2020) 4335–4385.

[23] F. Yang, C. Du and L. Huang, Ensemble Sentiment Analysis Method based on R-CNN and C-RNN with Fusion Gate, International Journal of Computers Communications & Control. 14 (2019) 272–285.

[24] S. Yang, X. Yu and Y. Zhou, LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example, in: 2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI), IEEE, Shanghai, China, (2020) pp. 98–101.

[25] L. Zhang, S. Wang and B. Liu, Deep learning for sentiment analysis: a survey, data mining and knowledge dscovery. 8 (2018) 1–34.

[26] D.-X. Zhou, Universality of Deep Convolutional Neural, Applied and Computational Harmonic Analysis. 48 (2020) 787–794.

[27] X. Zhou, X. Wan and J. Xiao, Attention-based LSTM Network for Cross-Lingual Sentiment Classification, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, Texas, (2016) pp. 247–256.

[28] SemEval2016 task5, BhMad Studio. (2016). Available: http://alt.qci.org/semeval2016/task5

[29] SemEval2014 task4 (2014). Available: http://alt.qcri.org/semeval2014/task4/