A Deep Learning Approach to Sarcasm Detection from Composite Textual Data

AMIT KHAN[1]
DIPANKAR MAJUMDAR[1]
BIKROMADITTYA MONDAL[2]
SOUMEN MUKHERJEE[1]

RCC Institute of Information Technology[1]
Canal South Road, Beliaghata, Kolkata-15, India[1]
B.P. Poddar Institute of Management and Technology, [2]
137, V.I.P. Road, Kolkata - 52, India[2]
[1] (amit.it.khan,dipankar.majumdar,soumou601)@gmail.com
[2] bikmondal@gmail.com

Abstract. Sarcasm is a means of conveying a bad attitude through social media platforms by utilizing positive or exaggerated positive terms. The last decade witnessed sarcasm detection to become a highly phenomenal topic of research; however the task of automated detection of sarcastic comments in a text remains an elusive problem. Sarcasm detection has eventually become a considerably significant task in the domain of sentiment classification. Without properly detecting the sarcasm from textual comments, sentiment classification remains incomplete and may lead to wrongful conclusion and decision. In this paper, we present a recurrent neural network (RNN)-based bidirectional long-short term memory (Bi-LSTM) network for sarcasm detection.The proposed technique has been applied to a combined dataset which is produced form news headline sarcasm dataset and news headline sarcasm version 2. Results of our technique renders enhanced performance over the existing technique found in literature.

Keywords: Bi-LSTM; Sarcasm; Deep Learning; RNN; Sentiment Analysis.

(Received November 19th, 2022 / Accepted December 1st, 2022)

1 Introduction

In recent times, Natural Language Processing (NLP) [1] has emerged as a highly significant and dynamic field for research and development.It allows computers to 'understand' [2] language of human beings in the form of either textual or speech data, including the speaker's or writer's intent and sentiment. NLP aims at automating [3] the task of recognizing linguistic text or voice and respond to them in real time. While watching online chats, sentiment analysis is a process of text mining [3] that discovers and extracts subjective information in source material, assisting a corporate organisation in understanding the societal opinion of their product, commodities, or service [4–7]. Several studies can be performed using opinion mining. These analytics may be used to find and fetch various levels of opinion. The analysis process is examined along several individual entities i.e., by words or phrases in the document. Sentimental words [8] can have either a positive or negative connotation [9–11]. There are several challenges confronted by Opinion Mining [12, 13] such as Named Entity Recognition [14, 15], Anaphora Recognition, parsing, sarcasm detection and many more [12, 16, 17].

Many people now days express [18] their opinions on different social websites. People have started to specify their opinion in sarcasm. Sarcasm is frequently thought of as an individual's opinion on extremely complex issues with very precise aims. One of the key states of affairs is the detection of sarcasm [8, 19] faced in Opinion Mining. Sarcasm detection [20] also known as irony detection is the way of transferring opinion in indirect manner. It is essentially an unbearable reflection which is conveyed through negating the [21] contextual meaning, can also indicate a state of ambivalence.

In NLP, sarcasm or existence of sarcastic phrases/comments often poses substantial hindrances. Sarcasm is defined as linguistic irony that displays negative and critical attitudes about people or events [22]. Consequently, it becomes a challenging task to identify and extract the sarcastic words, phrases or sentences from the text, before we can process them for sentiment analysis. Sarcasm can be stated in different ways.It can be stated verbally [20] or in writing [23]. Sarcasm can be expressed in a variety of forms, including direct dialogue, speech, and text etc. In case of direct conversation, facial expression and body movements give the clue of sarcasm. In the speech, sarcasm can be inferred if there is any alteration in tone of voice. On the other hand in text, it is hard to determine sarcasm [19, 20] compared to other techniques, but it can be conveyed [24] using a block letter, high use of exclamatory signs, hyperbole, utilization of emoticons and many more.

Most of the former works [20, 23, 24] for detecting sarcasm were mainly concentrated on rule based and statistical approaches using various features like lexical, pragmatic, existence of punctuation, opinion shift, intensifier etc.

Effective performance of deep learning model in various NLP tasks has been reported in [25] like text summarization, machine translation and question answering, and even in sarcasm detection [26, 27], Taking cue from the above works reported in [20, 23, 24] this paper attempts to identify the sarcastic comments in a textual dataset. The results of the work are expected to sieve out the sarcastic words/phrases from the dataset making it suitable for using on conventional sentiment analysis tools and platforms. With reference to proven utility of the technology in the domain in [28–30], we have used RNN and its advanced version are specially built for textual data analysis. Motivated with the elegance and success of works presented in [31–37], we have used the Bi-LSTM [38–40] network to detect sarcastic news headlines from our dataset.

The following is how the article is structured. Section 2 contains related works, followed by proposed method in section 3, result and discussion in section 4 and conclusion and future direction of the work in section 5.

2 Related Works

Many scholars have presented numerous approaches to identifying sarcasm clues in a particular text utilizing various rule-based and statistics-based methodologies in the literature.

In the domain of generic sentiment analysis various works such as [8, 22] have been noticed. However authors in the papers [8,19,21] have reported that sarcasm becomes a major point of concern in the analysis of opinions. Consequently, works like [8, 20, 22, 23] have reported significant contributions in the domain of sarcasm identification and extraction. Works like [19, 24] reports usage and identification of sarcastic emoticons in textual comments.

Thereafter, the objectives have been realized using Deep Learning based techniques [26–28, 31]. Out of the prevailing technologies, RNN and its implementation as Long Short-Term Memory (LSTM) and its allied approaches has emerged [29, 30, 32, 33, 35–37] out as the major trustworthy solution in the domain.

In [41] Tsur et al. tried to find sarcastic reviews from Amazonâs product reviews dataset using different types of features like syntactic based, patterns as well as punctuation based. Their approach was semi-supervised type. Sriram et al. [42] classified tweets into a preset set of generic groups including events, opinions, deals, and secret messages using non-contextual criteria such as the existence of slangs, time-event expression, opinion oriented words, and individual Twitter information. In [43] Lukin et al. applied a semi-supervised method using boot-strapping technique.

In [44] the authors proposed method using features like uni-grams and other pragmatic types of features. In [45] Tungthamthiti et al. analyzed twitter data to find sarcasm using concept based sentiment classification approach mainly based on supervised learning mechanism. Riloff et al. showed in [21] that the existence of good sentiment in a bad scenario can be used to detect sarcasm. In [46] Rajadesingan et al. applied their effort to construct a model that was able to detect sarcasm from the textual data. They proposed a strategy that was based on behavioural and psychological research.

In [47], Wang et al. recommended that the context modeling should be more important rather than detecting whether a tweet is sarcastic or not. As per their opinion, based on the context the result may be different and finally, they used a sequential classification task to model the problem. In [48] Amir et al. used a technique to automatically extract relevant features from tweets using convolutional neural network (CNN) based deep learning approach and augment them with user embeddings to provide more contextual features during sarcasm detection. Ghosh and Veale suggested a sarcasm detection system based on a deep neural network Â that stacked a CNN on top of LSTM in [49]. Â Bouazizi and Otsuki employed a pattern-based approach to detect irony from Tweets in [50].

Muresan et al. proposed a technique in [51] for constructing a collection of sarcastic Twitter messages in which the tweet writer provides the tagging message of the tweet, i.e. sarcastic or non-sarcastic. They looked at how the machine learning algorithm's performance fluctuates depending on the lexical and pragmatic components of the messages to determine and uncover sarcastic tweets, and they prioritized the features based on their contribution to the categorization.

Zhang et al. [52] proposed a study in which they attempted to assess sarcasm in a tweet published on Twitter using contextual elements of the tweet itself. The challenge is not limited to this; in [53], Felbo et al. attempted to develop a new framework for analysing phrases to extract sentence-specific sentiment using neural networks, which they then used to detect sarcasm and personality traits in any person. Ghosh and Muresan used the LSTM algorithm to determine irony in chat communications in [54]. Hazarika et al. [55] describe a fusion approach in which they extract contextual messages from a discussion thread's discourse section and use user embeddings to encode users' varied qualities such as stylometric and personality features. On a huge Reddit1 corpus, their sarcasm detection model shows encouraging results.

Zhang et al. proposed a model for semantic categorization of phrases that combined both convolution neural networks and bi-directional gated recurrent units in [56]. Han et al. employed CNN and Bi-LSTM to create a text-based sentiment categorization approach that utilized four different memory network models in [57].

CNN and RNN in mixed mode have shown outstanding performance in natural language processing, according to MohdUsama et al. in [34]. They suggested a firsthand model based on RNN with a CNN-based attention mechanism, combining the benefits of both architectures in one model, and empirically validating it.

In this work we have used Bi-LSTM network which provides a significant performance improvement compared to the earlier works in this area.

3 Proposed Method

The proposed work addresses the problem of sarcasm detection and categorizing the same as to whether the text is either sarcastic or not. The implementation is carried out using Bi-LSTM model of RNN. The input layer relies on Keras word embeddings using predefined vocabulary. During training, the embedding weights are adjusted iteratively. 64 output neurons were used to model the core of the constructed Bi-LSTM network. After that, the output is combined into a single drop-out layer. For classification, the output from the drop-out layer is transferred to the final sigmoid layer. The classification is obtained in the form of binary predictions. The network has been trained with RMSprop optimizer using binary cross entropy loss function. The drop-out layer accomplishes the regularization of the network with a high drop-out rate of 0.5. The network uses early stopping regularization mechanism with patience value: 5, in respect to minimum validation loss.

3.1 Model structure and different hyper parameters

Figure 1 depicts the essential framework of our suggested model. Different layers used in our model are discussed subsequently.

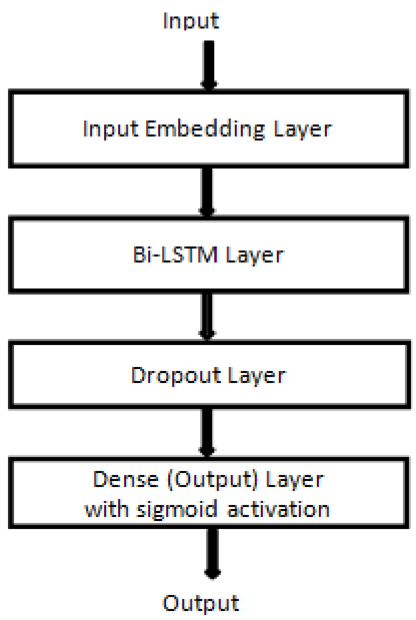

Figure 1: Model structure of our proposed system

3.1.1 Input embedding layer

We have used Keras Embedding at the input embedding layer for network. It requires an integer encoded input, where each word is represented by a distinct integer. Initially random weights were assigned to the embedding layer. Subsequently with exhaustive training, they are optimized to a desired level .For every of the words in the training dataset, the network learns embedding. This embedding layer acts as the first hidden layer of the network. After repeated empirical analysis, we have arrived to the following values of the critical parameters for our proposed model that gave the best results.

- Input dimension: This parameter refers to the vocabulary size of the input text data. In this case, the size of the input dimension is 15000.
- Output dimension: This parameter refers to the vector space size in which words will be embedded. It specifies the length of the output vectors for each and individual word from this layer. We have used 100 as the output vector size.
- Input length: This is the length of input sequences for any input layer of a Keras model. It indicates maximum number of words in a textual document or a comment. For the proposed work it amounts to 60. This paper assumes each news headline to contain a maximum of 60 words. If number of words of a particular headline is more or less than 60 then truncation or padding will be done as per necessity. Truncation or padding may be pre or post depending upon the case. The proposed work uses post truncation/padding.

3.1.2 Bi-LSTM layer

In this work we have used a Bidirectional LSTM hidden layer with 64 memory cells. Here total number of memory cells is 128. Out of these, 64 cells have been used for the transmission in the forward direction and another 64 for the transmission in the backward direction. The current work employs an exceeding number of memory cells and a deeper network expecting an achievement of better results. A systematic view of the Bidirectional LSTM is shown in figure 2.

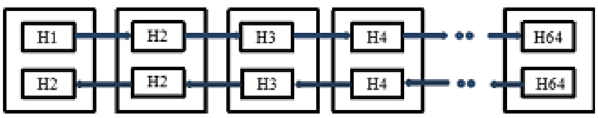

Figure 2: Bidirectional LSTM

The LSTM network utilized in this study was designed to improve learning and memory of long-term dependencies while avoiding vanishing gradient descent and explosion problems, may be conventionally modeled using the following system of equations. The proposed model typically consists of three gates and a cell memory state. Each of the cell state may be computed as follows:

$$X = \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \quad (1)$$

$$X = \sigma(W_f.X + b_f) \quad (2)$$

$$X = \sigma(W_i.X + b_i) \quad (3)$$

$$o_t = \sigma(W_i.X + b_i) \quad (4)$$

$$c_t = f_t \bigodot c_{t-1} + i_t \bigodot tanh(W_c.X + b_c) \quad (5)$$

$$h_t = o_t \bigodot tanh(c_t) \quad (6)$$

Where the symbols used represents the following:

i$_t$: $InputGate$
f_t : $Forget\ Gate$
o_t : $Output\ Gate$
c_t : $Cell\ State$
$W_i, W_f, W_o \in R^{d\times 2d}$: $Weighted\ Matrices\ used\ for\ mapping\ the\ Hidden\ Layer\ Input\ to\ the\ Three\ Gates\ and\ the\ Input\ Cell\ State.$
$b_i, b_f, b_o \in R^d$: $Biases$
σ : $Sigmoid\ Function\ and\ stands\ for\ element\ wise\ multiplication$
x_t, h_t : $Input\ Vector\ and\ Hidden\ Vector\ of\ LSTM\ Cell\ Unit$

The Bi-LSTM employed in our system is typically composed a forward and a backward LSTM layer. The forward layer keeps track of the past archaic information in forward sequence. On the other hand, the backward layer does just the reverse. It is responsible for studying and learning related dependencies in reverse sequence. Both layers are used for learning as well as identification of sarcastic features and characteristics from an input sequence. Consequently both are connected to the same output layer.

3.1.3 Dropout layer

We have characteristically used a dropout layer following the bidirectional LSTM layer. This is mainly used for regularization purpose. Here we have used a high dropout rate of 0.5 to overcome over fitting.

3.1.4 Dense output layer

Since our problem is purely a binary classification problem, only one neuron has been used in the output layer with sigmoid activation function. The output of the sigmoid corresponds to its confidence in deciding whether the headline is sarcastic or not. The significant parameters of the output layer of our model have been elaborated as follows:

- Batch size: During training the model we have used the batch size as 128. We have tried using different batch size, but we found batch size 128 leads best result for our model.

- No.of Epochs: We have used 25 epochs to train the model and used early stopping regularization mechanism with patience 5 to resolve the problem of model over fitting. This means that after validation loss begins to decline, we will allow training to continue for up to a further 5 epochs, providing the training process an opportunity to overcome flat areas or find some additional improvement.

  A brief tabular representation of the above model with the tuned hyperparameters shown in Table1.

Table 1: Hyper parameters used in our proposed Model

Hyper Parameters	Values
Input Features (Vocabulary Size)	15000
Output Vector Dimension	100
Number of LSTM Units	128
Max Len of Input sentence (words)	60
Batch Size	64
No. of Epochs	25
Activation Function (Output Layer)	Sigmoid
Loss Function Used	Binary cross entropy
Optimizer	RMSprop

4 Result and Discussion

The proposed approach has been executed on combined dataset of News headlines for Sarcasm and News headlines for Sarcasm v2 from Kaggle. In the said approach 80 % of the dataset has been used for training and rest 20 % for testing. The model has been trained with 25 epochs and at the same time dropout and early stopping regularization techniques has been applied with patience value set to 5. A comparative analysis of the proposed approach along with the other contemporary approaches has been presented in Table 2. A sharp rise of 8 % in accuracy of the presented approach in comparison with other approaches can be noticed thereby proving its novelity. Table 3 presents the details of the parameterized performance outcomes of the proposed model. This work achieves a precision-value of 0.972855, recall-value of 0.983246 and F1-Score of 0.978023. F1-Score is a highly vital and significant metric because it uses the precision and recall parameters together in a single metric. The proposed model with its hyper-parameters as presented in Table 1, achieves an accuracy of 99.15 % in training and 98.08 % in validation as graphically presented in Figure 3. Figure 4 graphically depicts the training and validation losses of 0.0303, and 0.0561 respectively are incurred in the execution of the model. The tabulated results of accuracy and losses attained in the model has been presented in Table 4.

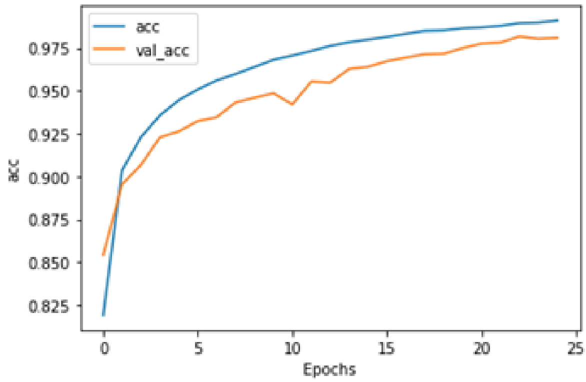

Figure 3: Training Accuracy Vs Validation Accuracy

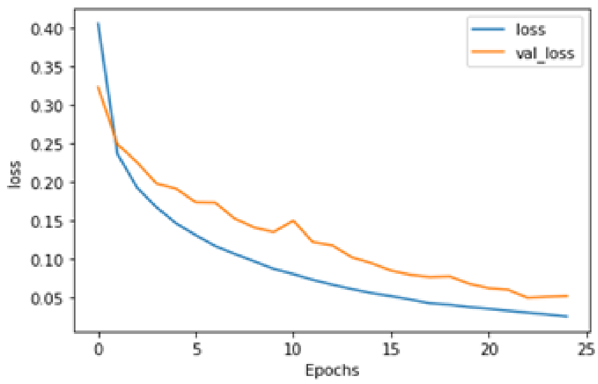

Figure 4: Training Accuracy Vs Validation Accuracy

Table 2: Performance comparison of our proposed model based on accuracy

Methods	Accuracy	Dataset
HNN [58]	89.70%	a
Deep CNN-LSTM [51]	86.16%	a
DDSD [59]	88.67%	a
[Proposed]	98.08%	b

Acronyms/Symbols used in Table - 2:

- HNN: Hybrid Neural Network
- DDSD: Dense and Deep Sarcasm Detection
- a: News headlines dataset for Sarcasm Detection from Kaggle
- b: Combination of News Headline for Sarcasm detection version1 and version2 dataset from Kaggle

Table 3: Other performance metrics of our proposed method

Precision	Recall	F1 Score
0.972855	0.983246	0.978023

Table 4: Performance metric of training and validation

Accuracy/Loss	Values
Training Accuracy	99.15%
Validation Accuracy	98.08%
Training Loss	0.0303
Validation Loss	0.0561

5 Conclusion and Future Scope

In this work we have explored and successfully established a novel approach for the utilization of Bi-LSTM Network based sarcasm identification technique on a combination of multiple dataset of news headlines obtained from www.kaggle.com. The different hyper parameters of model have been tuned in best possible way to achieve good accuracy. Our work claims to have achieved a very high accuracy nearly 98 %. In near future we wish to add other hybrid mechanisms on the top of Bi-LSTM to obtain better results.

References

[1] Manning, C. and Schutze, H. *Foundations of statistical natural language processing*. MIT press, 1999.

[2] Jurafsky, D. and Martin, J. H. Speech and language processing (draft). *preparation [cited 2020 June 1] Available from: https://web. stanford. edu/~jurafsky/slp3*, 2018.

[3] Zhao, J., Liu, K., and Xu, L. Sentiment analysis: mining opinions, sentiments, and emotions, 2016.

[4] Rodríguez, D. Z., Rosa, R. L., and Bressan, G. A billing system model for voice call service in cellular networks based on voice quality. In *2013 IEEE International Symposium on Consumer Electronics (ISCE)*, pages 89–90. IEEE, 2013.

[5] Nunes, R. D., Pereira, C. H., Rosa, R. L., and Rodríguez, D. Z. Real-time evaluation of speech quality in mobile communication services. In *2016 IEEE International Conference on Consumer Electronics (ICCE)*, pages 389–390. IEEE, 2016.

[6] de Almeida, F. L., Rosa, R. L., and Rodríguez, D. Z. Voice quality assessment in communication services using deep learning. In *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, pages 1–6. IEEE, 2018.

[7] Rodríguez, D. Z., Carrillo, D., Ramírez, M. A., Nardelli, P. H. J., and Möller, S. Incorporating wireless communication parameters into the e-model algorithm. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:956–968, 2021.

[8] Liu, B. *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge university press, 2020.

[9] Terra Vieira, S., Lopes Rosa, R., Zegarra Rodríguez, D., Arjona Ramírez, M., Saadi, M., and Wuttisittikulkij, L. Q-meter: Quality monitoring system for telecommunication services based on sentiment analysis using deep learning. *Sensors*, 21(5):1880, 2021.

[10] Lasmar, E. L., de Paula, F. O., Rosa, R. L., Abrahao, J. I., and Rodríguez, D. Z. Rsrs: Ridesharing recommendation system based on social networks to improve the userâs qoe. *IEEE Transactions on Intelligent Transportation Systems*, 20(12):4728–4740, 2019.

[11] Rosa, R. L., Rodríguez, D. Z., and Bressan, G. Music recommendation system based on user's sentiments extracted from social networks.

IEEE Transactions on Consumer Electronics, 61(3):359–367, 2015.

[12] Rodríguez, D. Z., da Silva, M. J., Silva, F. J. M., and Junior, L. C. B. Assessment of transmitted speech signal degradations in rician and rayleigh channel models. *INFOCOMP Journal of Computer Science*, 17(2):23–31, 2018.

[13] Vieira, S. T., Rosa, R. L., and Rodríguez, D. Z. A speech quality classifier based on tree-cnn algorithm that considers network degradations. *Journal of Communications Software and Systems*, 16(2):180–187, 2020.

[14] Jordane da Silva, M., Carrillo Melgarejo, D., Lopes Rosa, R., and Zegarra Rodríguez, D. Speech quality classifier model based on dbn that considers atmospheric phenomena. *Journal of Communications Software and Systems*, 16(1):75–84, 2020.

[15] Nunes, R. D., Rosa, R. L., and Rodríguez, D. Z. Performance improvement of a non-intrusive voice quality metric in lossy networks. *IET Communications*, 13(20):3401–3408, 2019.

[16] Rodríguez, D. Z., Rosa, R. L., Almeida, F. L., Mittag, G., and Möller, S. Speech quality assessment in wireless communications with mimo systems using a parametric model. *IEEE Access*, 7:35719–35730, 2019.

[17] Rodríguez, D. Z. and Junior, L. C. B. Determining a non-intrusive voice quality model using machine learning and signal analysis in time. *INFOCOMP Journal of Computer Science*, 18(2), 2019.

[18] Zarrella, D. *The social media marketing book*. " O'Reilly Media, Inc.", 2009.

[19] Gupta, S., Singh, R., and Singla, V. Emoticon and text sarcasm detection in sentiment analysis. In *First International Conference on Sustainable Technologies for Computational Intelligence*, pages 1–10. Springer, 2020.

[20] Yaghoobian, H., Arabnia, H. R., and Rasheed, K. Sarcasm detection: A comparative study. *CoRR*, abs/2107.02276, 2021.

[21] Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., and Huang, R. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 704–714, 2013.

[22] Kreuz, R. J. and Glucksberg, S. How to be sarcastic: The echoic reminder theory of verbal irony. *Journal of experimental psychology: General*, 118(4):374, 1989.

[23] Thakur, S., Singh, S., and Singh, M. Detecting sarcasm in text. In *International Conference on Intelligent Systems Design and Applications*, pages 996–1005. Springer, 2018.

[24] Castro, S., Hazarika, D., Pérez-Rosas, V., Zimmermann, R., Mihalcea, R., and Poria, S. Towards multimodal sarcasm detection (an _obviously_ perfect paper). *arXiv preprint arXiv:1906.01815*, 2019.

[25] Young, H. Young t., hazarika d., poria s., cambria e. *Recent trends in deep learning based natural language processing, IEEE Computational Intelligence magazine*, 13(3):55–75, 2018.

[26] Eke, C. I., Norman, A. A., and Shuib, L. Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and bert model. *IEEE Access*, 9:48501–48518, 2021.

[27] Kulkarni, S. and Biswas, A. Sarcasm detection methods in deep learning: Literature review. *ICT Analysis and Applications*, pages 507–512, 2020.

[28] Alharbi, N. M., Alghamdi, N. S., Alkhammash, E. H., and Al Amri, J. F. Evaluation of sentiment analysis via word embedding and rnn variants for amazon online reviews. *Mathematical Problems in Engineering*, 2021, 2021.

[29] Mahajan, D. and Chaudhary, D. K. Sentiment analysis using rnn and google translator. In *2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 798–802. IEEE, 2018.

[30] Saha, B. N., Senapati, A., and Mahajan, A. Lstm based deep rnn architecture for election sentiment analysis from bengali newspaper. In *2020 International Conference on Computational Performance Evaluation (ComPE)*, pages 564–569, 2020.

[31] Dhola, K. and Saradva, M. A comparative evaluation of traditional machine learning and deep learning classification techniques for sentiment analysis. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 932–936. IEEE, 2021.

[32] Monika, R., Deivalakshmi, S., and Janet, B. Sentiment analysis of us airlines tweets using lstm/rnn. In *2019 IEEE 9th International Conference on Advanced Computing (IACC)*, pages 92–95. IEEE, 2019.

[33] Ni, R. and Cao, H. Sentiment analysis based on glove and lstm-gru. In *2020 39th Chinese Control Conference (CCC)*, pages 7492–7497. IEEE, 2020.

[34] Usama, M., Ahmad, B., Song, E., Hossain, M. S., Alrashoud, M., and Muhammad, G. Attention-based sentiment analysis using convolutional and recurrent neural network. *Future Generation Computer Systems*, 113:571–578, 2020.

[35] Vimali, J. and Murugan, S. A text based sentiment analysis model using bi-directional lstm networks. In *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, pages 1652–1658. IEEE, 2021.

[36] Wen, S., Wei, H., Yang, Y., Guo, Z., Zeng, Z., Huang, T., and Chen, Y. Memristive lstm network for sentiment analysis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(3):1794–1804, 2019.

[37] Yenter, A. and Verma, A. Deep cnn-lstm with combined kernels from multiple branches for imdb review sentiment analysis. In *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pages 540–546. IEEE, 2017.

[38] Abduljabbar, R. L., Dia, H., and Tsai, P.-W. Unidirectional and bidirectional lstm models for short-term traffic prediction. *Journal of Advanced Transportation*, 2021, 2021.

[39] Goldberg, Y. Neural network methods for natural language processing. *Synthesis lectures on human language technologies*, 10(1):1–309, 2017.

[40] Zou, H., Wu, Y., Zhang, H., and Zhan, Y. Short-term traffic flow prediction based on pcc-bilstm. In *2020 International Conference on Computer Engineering and Application (ICCEA)*, pages 489–493. IEEE, 2020.

[41] Tsur, O., Davidov, D., and Rappoport, A. Icwsmâa great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In *fourth international AAAI conference on weblogs and social media*, 2010.

[42] Sriram, B., Fuhry, D., Demir, E., Ferhatosmanoglu, H., and Demirbas, M. Short text classification in twitter to improve information filtering. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 841–842, 2010.

[43] Lukin, S. and Walker, M. Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. *arXiv preprint arXiv:1708.08572*, 2017.

[44] González-Ibánez, R., Muresan, S., and Wacholder, N. Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586, 2011.

[45] Tungthamthiti, P., Shirai, K., and Mohd, M. Recognition of sarcasms in tweets based on concept level sentiment analysis and supervised learning approaches. In *Proceedings of the 28th Pacific Asia conference on language, information and computing*, pages 404–413, 2014.

[46] Rajadesingan, A., Zafarani, R., and Liu, H. Sarcasm detection on twitter: A behavioral modeling approach. In *Proceedings of the eighth ACM international conference on web search and data mining*, pages 97–106, 2015.

[47] Wang, Z., Wu, Z., Wang, R., and Ren, Y. Twitter sarcasm detection exploiting a context-based model. In *international conference on web information systems engineering*, pages 77–91. Springer, 2015.

[48] Amir, S., Wallace, B. C., Lyu, H., and Silva, P. C. M. J. Modelling context with user embeddings for sarcasm detection in social media. *arXiv preprint arXiv:1607.00976*, 2016.

[49] Ghosh, A. and Veale, T. Fracking sarcasm using neural network. In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 161–169, 2016.

[50] Bouazizi, M. and Ohtsuki, T. O. A pattern-based approach for sarcasm detection on twitter. *IEEE Access*, 4:5477–5488, 2016.

[51] Muresan, S., Gonzalez-Ibanez, R., Ghosh, D., and Wacholder, N. Identification of nonliteral language in social media: A case study on sarcasm.

Journal of the Association for Information Science and Technology, 67(11):2725–2737, 2016.

[52] Zhang, M., Zhang, Y., and Fu, G. Tweet sarcasm detection using deep neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: technical papers*, pages 2449–2460, 2016.

[53] Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., and Lehmann, S. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *arXiv preprint arXiv:1708.00524*, 2017.

[54] Ghosh, D. and Muresan, S. " with 1 follower i must be awesome: P." exploring the role of irony markers in irony recognition. In *Twelfth International AAAI Conference on Web and Social Media*, 2018.

[55] Hazarika, D., Poria, S., Gorantla, S., Cambria, E., Zimmermann, R., and Mihalcea, R. Cascade: Contextual sarcasm detection in online discussion forums. *arXiv preprint arXiv:1805.06413*, 2018.

[56] Zhang, D., Tian, L., Hong, M., Han, F., Ren, Y., and Chen, Y. Combining convolution neural network and bidirectional gated recurrent unit for sentence semantic classification. *IEEE access*, 6:73750–73759, 2018.

[57] Han, H., Liu, J., and Liu, G. Attention-based memory network for text sentiment classification. *IEEE Access*, 6:68302–68310, 2018.

[58] Misra, R. and Arora, P. Sarcasm detection using hybrid neural network. *arXiv preprint arXiv:1908.07414*, 2019.

[59] Pelser, D. and Murrell, H. Deep and dense sarcasm detection. *arXiv preprint arXiv:1911.07474*, 2019.