

# Deep Learning with Natural Language Processing Enabled Sentimental Analysis on Sarcasm Classification

Abdul Rahaman Wahab Sait<sup>1,\*</sup> and Mohamad Khairi Ishak<sup>2</sup>

<sup>1</sup>Department of Documents and Archive, Center of Documents and Administrative Communication, King Faisal University, Al Hofuf, Al-Ahsa, 31982, Saudi Arabia

<sup>2</sup>School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia (USM), Nibong Tebal, Penang, 14300, Malaysia

\*Corresponding Author: Abdul Rahaman Wahab Sait. Email: [asait@kfu.edu.sa](mailto:asait@kfu.edu.sa)

Received: 07 March 2022; Accepted: 07 April 2022

**Abstract:** Sentiment analysis (SA) is the procedure of recognizing the emotions related to the data that exist in social networking. The existence of sarcasm in textual data is a major challenge in the efficiency of the SA. Earlier works on sarcasm detection on text utilize lexical as well as pragmatic cues namely interjection, punctuations, and sentiment shift that are vital indicators of sarcasm. With the advent of deep-learning, recent works, leveraging neural networks in learning lexical and contextual features, removing the need for handcrafted feature. In this aspect, this study designs a deep learning with natural language processing enabled SA (DLNLP-SA) technique for sarcasm classification. The proposed DLNLP-SA technique aims to detect and classify the occurrence of sarcasm in the input data. Besides, the DLNLP-SA technique holds various sub-processes namely preprocessing, feature vector conversion, and classification. Initially, the pre-processing is performed in diverse ways such as single character removal, multi-spaces removal, URL removal, stopword removal, and tokenization. Secondly, the transformation of feature vectors takes place using the N-gram feature vector technique. Finally, mayfly optimization (MFO) with multi-head self-attention based gated recurrent unit (MHSA-GRU) model is employed for the detection and classification of sarcasm. To verify the enhanced outcomes of the DLNLP-SA model, a comprehensive experimental investigation is performed on the News Headlines Dataset from Kaggle Repository and the results signified the supremacy over the existing approaches.

**Keywords:** Sentiment analysis; sarcasm detection; deep learning; natural language processing; n-grams; hyperparameter tuning

## 1 Introduction

Sarcasm is a rhetorical way of exposing dislike or negative emotion through exaggerated language construct. It is a variety of mockery and false politeness for intensifying hostility without plainly accordingly [1]. In face-to-face discussion, sarcasm is effortlessly identified through gestures, tone of the



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

speaker, and facial expressions. But identifying sarcasm in written messages is not an insignificant process since none of these cues is easily accessible [2]. Through internet, sarcasm recognition in online communication from discussion forums, e-commerce websites, and social media platforms has turned out to be critical for online trolls, sentiment analysis, identifying cyberbullies, and opinion mining [3]. The topic of sarcasm gained considerable attention from Neuropsychology to Linguistics [4]. During this case, the feature is handcrafted and could generalize in the incidence of figurative slang and informal language that is extensively utilized in online conversation [5].

Sarcasm Detection (SD) from Twitter is modelled as a binary text classification process [6]. Detecting Sarcasm in text classification is a major process with various suggestions for several fields namely fitness, safety, and marketing. The SD method could assist companies to examine customer sentiment regarding the good [7]. It leverages the company to promote the quality of the product. In sentimental analysis, classification of sentiment is the main subfunction, particularly to categorize twitters, comprising hidden data in the message that a person shares with others. Also, one could utilize the composition of Twitter for predicting sarcasm. Executing machine learning (ML) algorithm could produce effective outcomes for detecting sarcasm [8]. Constructing an efficient classification method based on several factors. The major factor is the attribute utilized and the sovereign attribute in the learning model that is effortlessly integrated into the class example [9]. With the emergence of deep learning (DL), current studies [10], leverage NNs for learning contextual and lexical features, eliminate the necessity for handcrafted features. When DL based methods accomplish remarkable results, it can be lack interpretability.

This study designs a deep learning with natural language processing enabled SA (DLNLP-SA) technique for sarcasm classification. The proposed DLNLP-SA technique holds various sub-processes namely pre-processing, feature vector conversion, and classification. Initially, the pre-processing is performed in diverse ways such as single character removal, multi-spaces removal, URL removal, stopword removal, and tokenization. Secondly, the transformation of feature vectors takes place using the N-gram feature vector technique. Finally, mayfly optimization (MFO) with multi-head self-attention based gated recurrent unit (MHSA-GRU) model is employed for the detection and classification of sarcasm. To verify the enhanced outcomes of the DLNLP-SA model, a comprehensive experimental investigation is performed on the benchmark dataset.

## 2 Related works

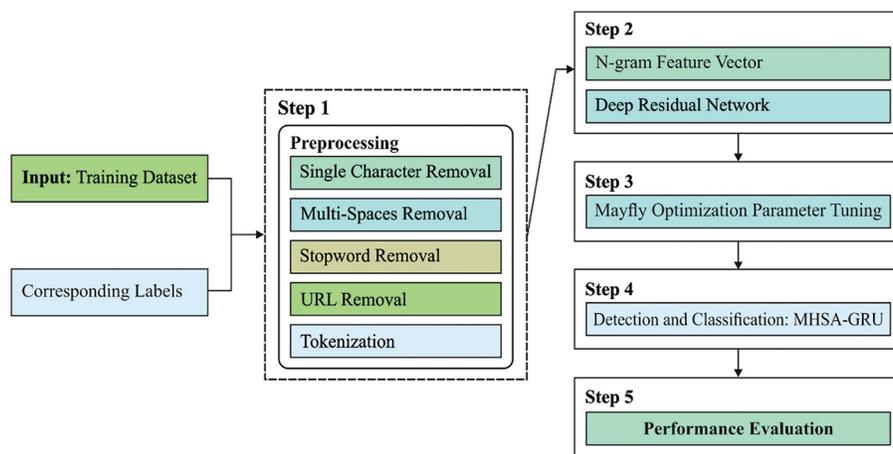
Wen et al. [11] presented a sememe and auxiliary improved attention neural method, SAAG. On the word level, it can be present sememe knowledge for enhancing the representation learning of Chinese words. The sememe is the minimal unit of meaning that is fine-grained representation of words. Bedi et al. [12] established a Hindi-English code-mixed data set, MaSaC for multi-modal sarcasm recognition and humor classifier from conversational dialog that to skill is primary data set of their kind; (2) it can be present MSH-COMICS, a new attention-rich neural infrastructure to utterance classifier. Ren et al. [13] introduced a multi-level memory network utilizing sentiment semantics (SS) for taking the feature of sarcasm expression. During this method, it utilizes the 1st-level memory network for taking SS and utilizes the 2nd-level memory network for taking the contrast amongst SS and the condition from all the sentences. In addition, it can be utilized an improved CNN for increasing the memory network from the absence of local information.

Zhang et al. [14] presented the complex-valued fuzzy network by leveraging the mathematical formalism of quantum model and fuzzy logic (FL). The contextual interface amongst neighboring utterances are explained as the interaction amongst a quantum model and its surrounding environment, creating the quantum composite method, whereas the weight of interface was defined as a fuzzy membership function. Nayak et al. [15] estimated different vectorization and ML techniques for detecting

sarcastic headlines. In experiments illustrate that pre-trained transformer based embedded integrated with LSTM network offer optimum outcomes. The author in [16] examined negative sentiment tweets with occurrence of hyperboles for SD. In 6000 and 600 preprocessing negative sentiment tweets containing #Kungflu, #Coronavirus, #Chinesevirus, #COVID19, and #Hantavirus are collected to SD. In 5 hyperbole features like capital letter, elongated word, interjection, intensifier, and punctuation mark are analyzed utilizing 3 well-known ML techniques.

### 3 The Proposed Model

In this study, a new DLNLP-SA technique aims for detecting and classifying the occurrence of sarcasm in the input data. The proposed DLNLP-SA technique undergoes distinct set of processes such as pre-processing, N-gram feature extraction, MHSA-GRU based classification, and MFO based hyperparameter optimization. Fig. 1 demonstrates the block diagram of DLNLP-SA technique.



**Figure 1:** Block diagram of DLNLP-SA technique

#### 3.1 Pre-Processing

When the dataset is recognized, a primary step is to pre-process the text data. The text pre-processed is the method of cleaning the new text data. The strong text pre-processed method is indispensable to application on NLP tasks. Since every textual element is attained then pre-processing serves as an important component of input that is offering text data application. The pre-process comprises distinct techniques to translate the novel text as to well-defined procedure: lemmatization, removing of stopwords, lexical analysis (removing of punctuations, special character or symbol, word tokenization, and ignore case sensitivity). The various sub processes limited in data pre-processed are:

- Removing numerals
- Removing stop words
- Removing multiple spaces
- Removing punctuation marks
- Removing single letter words and
- Change uppercase letters into lowercase.

### 3.2 N-Gram Feature Extraction

The pattern is created with the concatenation of neighboring tokens to n-grams whereas  $n = \{1, 2, 3, \dots\}$ . During the easy manner, in which,  $n$  refers the similar to one is named as unigram. Employing n-grams, it takes the possibility of taking that word to be possible to develop visible from the text in terms of other words. Usually,  $n$  - grams are certainly not larger than  $n = 3$ . The maximum values are the probability for generating complex pattern that equals rarely.

### 3.3 MHSA-GRU Based Sarcasm Classification

At the time of SD and classification, the MHSA-GRU model has been employed to it. The GRU offers a basic unit comprised of two control gates [17], reset gate and the update gate, rather than the absence of the cell state and three in the LSTM cell. The reset and update gates are parallel to forget gate an input gate from the LSTM cell however the variance is in how the output of this gate is utilized within the GRU cell. Consequently, a GRU cell has lesser training parameters when compared to LSTM, making the training very fast. Especially, GRU was proposed for capturing dependency of distinct time scales in machine translation tasks.

$$z_t = \sigma(W_z x_t + R_z h_{t-1} + b_z) \quad (1)$$

$$r_t = \sigma(W_r x_t + R_r h_{t-1} + b_r) \quad (2)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \psi(W_h x_t + R_h (r_t \odot h_{t-1}) + b_h) \quad (3)$$

whereas  $W_z, W_r, W_h \in \mathbb{R}^{N \times M}$ ,  $R_z, R_r, R_h \in \mathbb{R}^{N \times N}$  and  $b_z, b_r, b_h \in \mathbb{R}^N$  denotes weight matrices and bias vectors and  $x_t, z_t, r_t, h_t \in \mathbb{R}^N$  denotes input, update gate, reset gate, and output vector, correspondingly.

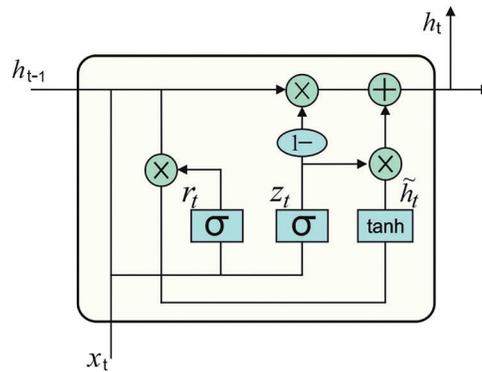
A GRU cell integrates the input and forgets gates of LSTM into one single update gate. This gate decides how much data from the preceding hidden state need to be passed through following hidden states. The reset gate  $r_t$  enables to drop data unrelated for the future. Subsequently, GRU simplifies LSTM by removing specific parameters however, it maintains its important property.

Assume a sentence  $S$ , we employ typical tokenizer and utilize *pre-train* ed methods for obtaining  $D$  dimension embedding for single word in the sentence. This embedding  $S = \{e_1, e_2, \dots, e_N\}$ ,  $S \in \mathbb{R}^{N \times D}$  from the input to this algorithm. For detecting sarcasm in sentence  $S$ , it is critical for identifying certain words that offer important cues namely sarcastic connotation and negative emotion. The significance of this cue-word is based on various aspects according to the distinct different contexts. In this presented method, we leverage multi-head self-attention model for recognizing this cue-word from the input text. Fig. 2 depicts the framework of GRU.

Attention mechanism finds pattern in the input that is critical for resolving the provided process. In DL method, self-attention [18] is an attention method for sequence that assists in learning the task-specific relationships among distinct components of a provided sequence for producing a good sequence representation. During the self-attention model, 3 linear predictions: Key (K), Value (V), and Query (Q) of the provided input sequence is produced in which,  $Q, V \in \mathbb{R}^{N \times D}$ .

$$A = \text{softmax} \left( \frac{QK^T}{\sqrt{D}} \right) V \quad (4)$$

In multiple-head self-attention, various duplicates of the self-attention model are parallely utilized. All the head takes distinct relations among the words in the input text and recognizes the keyword that assistance in classification.



**Figure 2:** Framework of GRU

### 3.4 MFO Based Hyperparameter Optimization

During hyperparameter tuning process, the MFO algorithm has been applied to properly tune the hyperparameters involved in it [19]. An optimization is utilized for determining the particular and maximum accurate solution to these problems. During this case, MO was utilized for finding particular and accurate solutions. The mayflies (MFs) are insects which are anciently named Palaeoptera. This MO is simulated as the social mate performance of MFs. The male mayflies (MMFs) appeal to female mayflies (FMs) by carrying out a nuptial dance on the water by creating up and down movements for procedure a design. This technique was generated by observing 3 procedures of these MFs and it can be movement of MMFs, movement of FMs, and mating method of MFs. These are described under:

As before mentioned the males, group, and dance on some meters of water. The MFs are incapable of moving at maximum speed and their velocity for reaching that level was computed by under in Eq. (6).

$$v_{ij}^{t+1} = v_{ij}^t + a_2 e^{-\beta r_p^2} (pbest_{ij} - x_{ij}^t) + a_1 e^{-\beta r_g^2} (pbest_{ij} - x_{ij}^t) \quad (5)$$

whereas,  $v_{ij}^t$  implies the velocity of MMF,  $x_{ij}^t$  refers the place of MMF,  $i = 1, \dots, n$  represents the space dimensional,  $t$  stands for the time step and  $i$  signifies the MF number. However,  $a_1$  and  $a_2$  are constants that are executed to scale for the contributions of social element and cognitive element. Besides,  $pbest_{ij}$  signifies the optimum place visited by MF,  $N$  denotes the entire amount of MMFs. Finally,  $\beta$  indicates the visibility co-efficient that border the visibility of MF to other MFs,  $r_p$  and  $r_g$  refers the space amongst and  $x_i$  and  $pbest_i$  and  $gbest$ .

$$\chi_i^{t+1} = X_i^t + v_i^{t+1} \quad (6)$$

$$v_i^{t+1} = v_{ij}^t + d \times r \quad (7)$$

The fresh place of MMF is estimated with Eq. (6) by summing up velocity  $v_i^{t+1}$  with existing place  $x_i^t$ . An optimum MFs retain on going with its nuptial dance by altering its velocity permitting to Eq. (7). Whereas,  $d$  denotes the co-efficient of nuptial dance and  $r$  implies the arbitrary number with range of  $-1$  and  $1$ .

The female does not swarm on its individual instead the FMs are concerned by males to breed. The magnetism procedure was demonstrated as method with determination as the velocity of all MFs both males as well as female are computed with employing its FF. Specifically, the FMF that is having maximum fitness value is magnetized by MMF is maximum fitness value. Therefore, the velocity of FMFs are computed as illustrated in Eq. (8):

$$v_i^{t+1} = \begin{cases} v_{ij}^t + a_3 e^{-\beta r_{mf}^2} (x_{ij}^t - y_{ij}^t) & \text{if } f(y_i) > f(x_i) \\ v_{ij}^t + fl * r & \text{if } f(y_i) \leq f(x_i) \end{cases} \quad (8)$$

In which,  $v_{ij}^t$  refers the velocity of FMF,  $y_{ij}^t$  represents the place of FMF,  $i = 1, \dots, n$  denotes the space dimensional,  $t$  implies the time step, and  $i$  stands for the MF number. In addition,  $a_3$  refers the constant practice for scaling the contribution of social elements and cognitive components. But,  $\beta$  signifies the visibility co-efficient, but  $r_{mf}$  refers the space amongst FMFs and MMFs.

$$y_i^{t+1} = y_i^t + v_i^{t+1} \quad (9)$$

Lastly,  $fl$  implies the co-efficient of arbitrary walk that is trained in event of the magnetization amongst FMF as well as MMF are failed, and  $r$  indicates the arbitrary number with  $-1$  and  $1$  range. A novel place  $y_i^{t+1}$  of FMF is established by adding their place for computing formula as in Eq. (9).

The procedure of mating amongst MMFs as well as FMFs are implemented with function is named crossover operator. As mentioned previously, fitness value was employed for choosing the partner for mating, and that outcomes in 2 offspring that are created as declared in Eqs. (10) and (11) under as offspring1 and offspring2.

$$offspring1 = L * male + (1 - L) * female \quad (10)$$

$$offspring2 = L * female + (1 - L) * male \quad (11)$$

In which,  $L$  represents the arbitrary number in the existing range, *male* imply the male parent and *female* signifies the female parent. Primary velocities of *offspring1* and *offspring2* are assumed that zero.

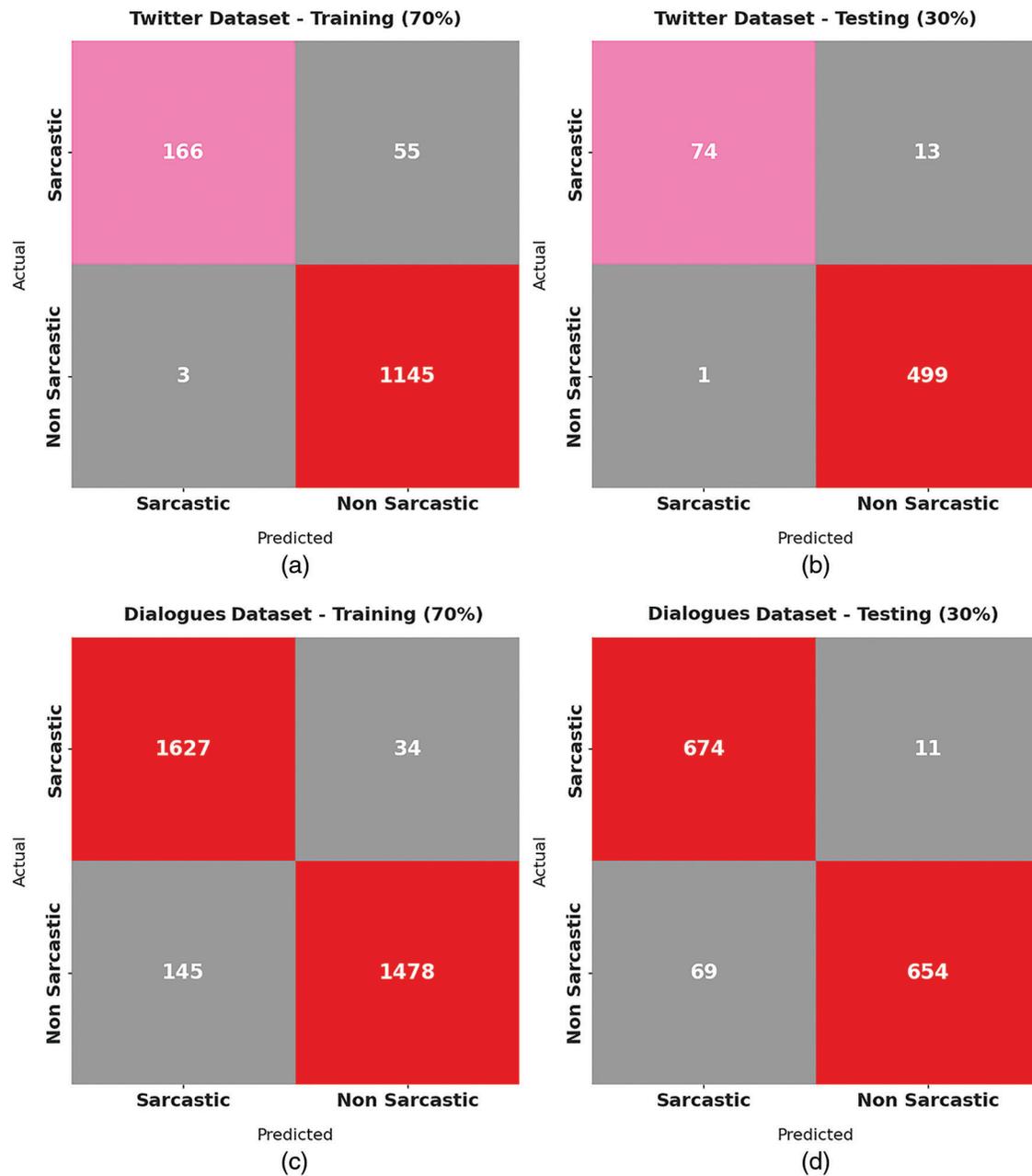
#### 4 Performance Validation

The performance of the DLNLP-SA model is validated using two benchmark datasets namely Twitter dataset and Dialogues dataset [20,21]. The first Twitter dataset includes 308 samples under sarcastic class and 1648 samples under non-sarcastic class. The latter Dialogues dataset includes 2346 samples into sarcastic class and 2346 samples into non-sarcastic class.

Fig. 3 offers a set of confusion matrices by the DLNLP-SA model on the training/testing data of Twitter dataset and Dialogues dataset. Fig. 3a indicates that the DLNLP-SA model has recognized 166 images into sarcastic class and 1145 images into non-sarcastic class on 70% of training data on Twitter dataset. Also, Fig. 3b represents that the DLNLP-SA algorithm has recognized 74 images into sarcastic class and 499 images into non-sarcastic class on 30% of testing data on Twitter dataset. Similarly, Fig. 3c refers that the DLNLP-SA method has recognized 1627 images into sarcastic class and 1478 images into non-sarcastic class on 70% of training data on Dialogue dataset. Likewise, Fig. 3d signifies that the DLNLP-SA methodology has recognized 674 images into sarcastic class and 654 images into non-sarcastic class on 30% of testing data on Dialogue dataset.

Tab. 1 provides detailed classification outcomes of the DLNLP-SA model on the Twitter dataset. The experimental results indicated that the DLNLP-SA model has resulted in effective outcomes on training as well as testing datasets.

Fig. 4 demonstrates the overall classification outcomes of the DLNLP-SA model on Twitter dataset. The DLNLP-SA model has classified the sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_t$ ,  $F_{score}$ , and MCC of 95.76%, 98.22%, 75.11%, 85.13%, and 83.72% respectively. In addition, the DLNLP-SA model has categorized the non-sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_t$ ,  $F_{score}$ , and MCC of 95.76%, 95.42%, 99.74%, 97.53%, and 83.72% respectively.

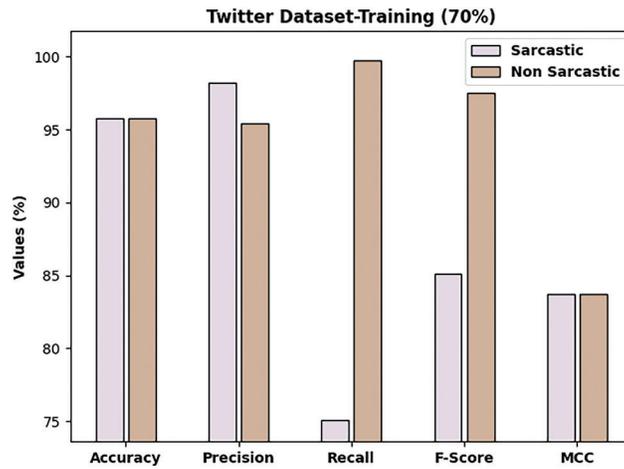


**Figure 3:** Confusion matrix of DLNLP-SA technique on training/testing data of Twitter dataset and Dialogues datasets

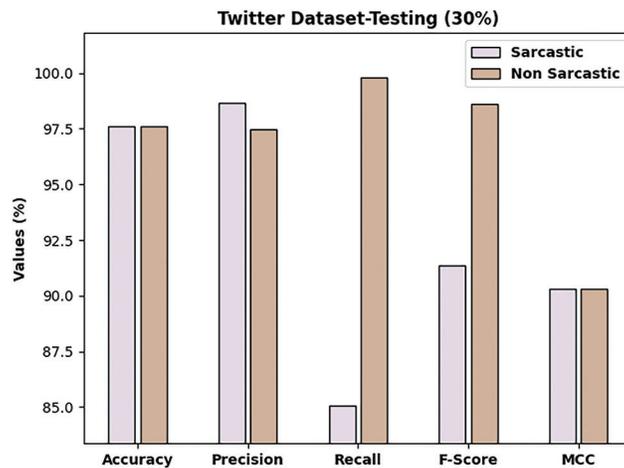
Fig. 5 showcases an overall classification outcomes of the DLNLP-SA model on Twitter dataset. The DLNLP-SA method has classified the sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_l$ ,  $F_{score}$ , and MCC of 97.61%, 98.67%, 85.06%, 91.36%, and 90.32% respectively. Eventually, the DLNLP-SA methodology has categorized the non-sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_l$ ,  $F_{score}$ , and MCC of 97.61%, 97.46%, 99.80%, 98.62%, and 90.32% correspondingly.

**Table 1:** Result analysis of DLNLP-SA model on the Twitter dataset

Twitter Dataset					
Class labels	Accuracy	Precision	Recall	F-Score	MCC
Training (70%)					
Sarcastic	95.76	98.22	75.11	85.13	83.72
Non Sarcastic	95.76	95.42	99.74	97.53	83.72
Average	95.76	96.82	87.43	91.33	83.72
Testing (30%)					
Sarcastic	97.61	98.67	85.06	91.36	90.32
Non Sarcastic	97.61	97.46	99.80	98.62	90.32
Average	97.61	98.06	92.43	94.99	90.32

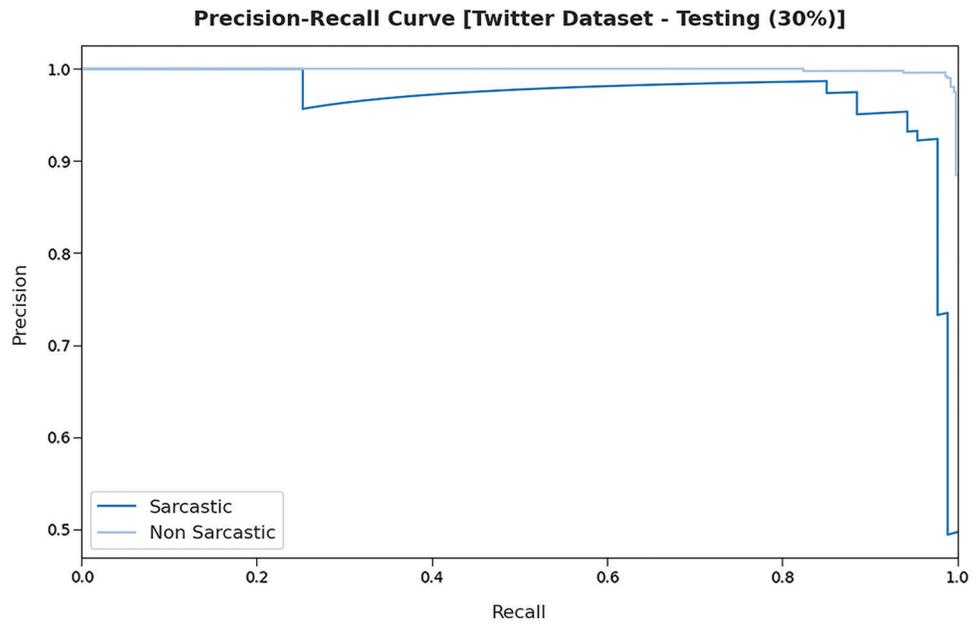


**Figure 4:** Result analysis of DLNLP-SA model on 70% of training data on Twitter dataset



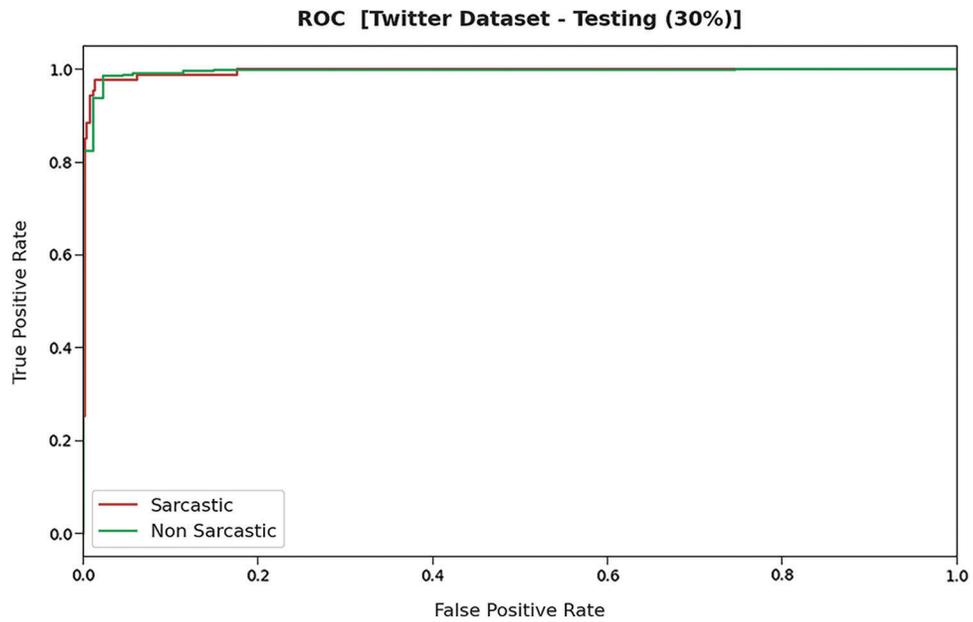
**Figure 5:** Result analysis of DLNLP-SA model on 30% of testing data on Twitter dataset

Fig. 6 reports the precision-recall curve analysis of the DLNLP-SA on the testing 30% of the Twitter dataset. The figures indicated that the DLNLP-SA model has resulted in effectual outcomes under Twitter dataset.



**Figure 6:** Precision-recall analysis of DLNLP-SA model on 30% of testing data on Twitter dataset

Fig. 7 demonstrates the ROC inspection of the DLNLP-SA model on the testing 30% of the Twitter dataset. The results indicated that the DLNLP-SA model has resulted in maximum performance on the testing dataset over the other ones.

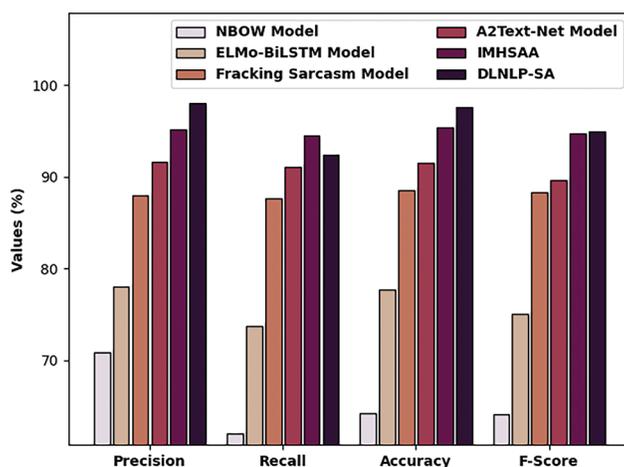


**Figure 7:** ROC analysis of DLNLP-SA model on 30% of testing data on Twitter dataset

A brief comparative study of the DLNLP-SA model with recent models on Twitter dataset is portrayed in Tab. 2 and Fig. 8. The experimental results indicated that the NBOW and ELMo-BiLSTM models have obtained lower classification outcomes over the other methods. At the same time, the Fracking sarcasm and A2Text-Net models have reached slightly improved performance. Followed by, the IMHSAA model has accomplished reasonable outcome. Finally, the proposed DLNLP-SA technique demonstrates the enhance result with  $prec_n$ ,  $reca_l$ ,  $accu_y$ , and  $F_{score}$  of 98.06%, 92.43%, 97.61%, and 94.99% respectively.

**Table 2:** Comparative analysis of DLNLP-SA technique with recent approaches on Twitter dataset

Methods	Precision	Recall	Accuracy	F-Score
NBOW Model	70.87	62.03	64.26	64.18
ELMo-BiLSTM Model	78.04	73.72	77.68	75.05
Fracking Sarcasm Model	88.04	87.63	88.57	88.29
A2Text-Net Model	91.59	91.06	91.49	89.65
IMHSAA	95.19	94.47	95.40	94.76
DLNLP-SA	98.06	92.43	97.61	94.99



**Figure 8:** Comparative analysis of DLNLP-SA technique on Twitter dataset

Tab. 3 offers detailed classification outcomes of the DLNLP-SA method on the Dialogues dataset. The experimental results indicated that the DLNLP-SA algorithm has resulted in effective outcomes on training as well as testing datasets.

Fig. 9 depicts an overall classification outcome of the DLNLP-SA approach on Dialogues dataset. The DLNLP-SA model has classified the sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_l$ ,  $F_{score}$ , and MCC of 94.55%, 91.82%, 97.95%, 94.79%, and 89.29% respectively. Moreover, the DLNLP-SA model has categorized the non-sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_l$ ,  $F_{score}$ , and MCC of 94.55%, 97.75%, 91.07%, 94.29%, and 89.29% correspondingly.

**Table 3:** Result analysis of DLNLP-SA model on the Dialogues dataset

Dialogues Dataset					
Class labels	Accuracy	Precision	Recall	F-Score	MCC
Training (70%)					
Sarcastic	94.55	91.82	97.95	94.79	89.29
Non Sarcastic	94.55	97.75	91.07	94.29	89.29
Average	94.55	94.78	94.51	94.54	89.29
Testing (30%)					
Sarcastic	94.32	90.71	98.39	94.40	88.95
Non Sarcastic	94.32	98.35	90.46	94.24	88.95
Average	94.32	94.53	94.43	94.32	88.95

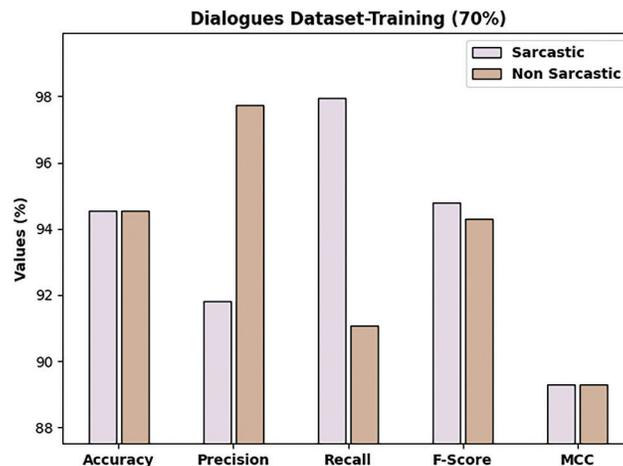
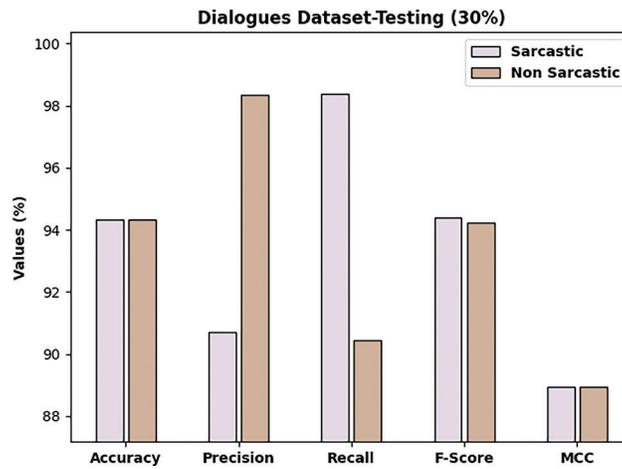
**Figure 9:** Result analysis of DLNLP-SA model on 70% of training data on Dialogues dataset

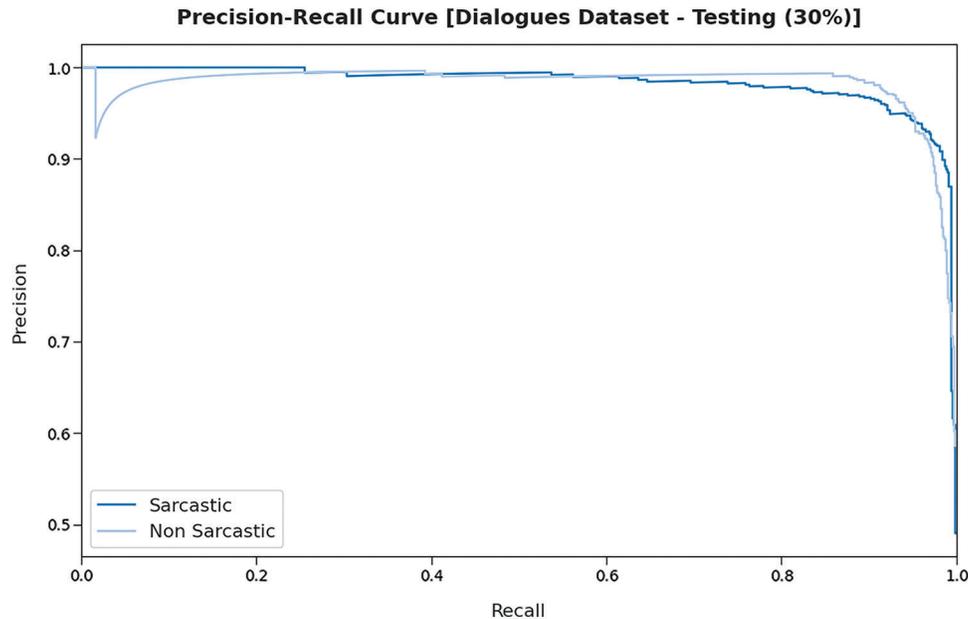
Fig. 10 illustrates the overall classification outcomes of the DLNLP-SA technique on Dialogues dataset. The DLNLP-SA model has classified the sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_l$ ,  $F_{score}$ , and MCC of 94.32%, 90.71%, 98.39%, 94.40%, and 88.95% correspondingly. Besides, the DLNLP-SA model has categorized the non-sarcastic samples with  $accu_y$ ,  $prec_n$ ,  $reca_l$ ,  $F_{score}$ , and MCC of 94.32%, 98.35%, 90.46%, 94.24%, and 88.95% respectively.

Fig. 11 demonstrates the precision-recall curve analysis of the DLNLP-SA system on the testing 30% of the Dialogues dataset. The figures indicated that the DLNLP-SA approach has resulted in effectual outcomes under Dialogues dataset.

Fig. 12 demonstrates the ROC inspection of the DLNLP-SA approach on the testing 30% of the Dialogues dataset. The results indicated that the DLNLP-SA technique has resulted to maximum performance on the testing dataset over the other ones.



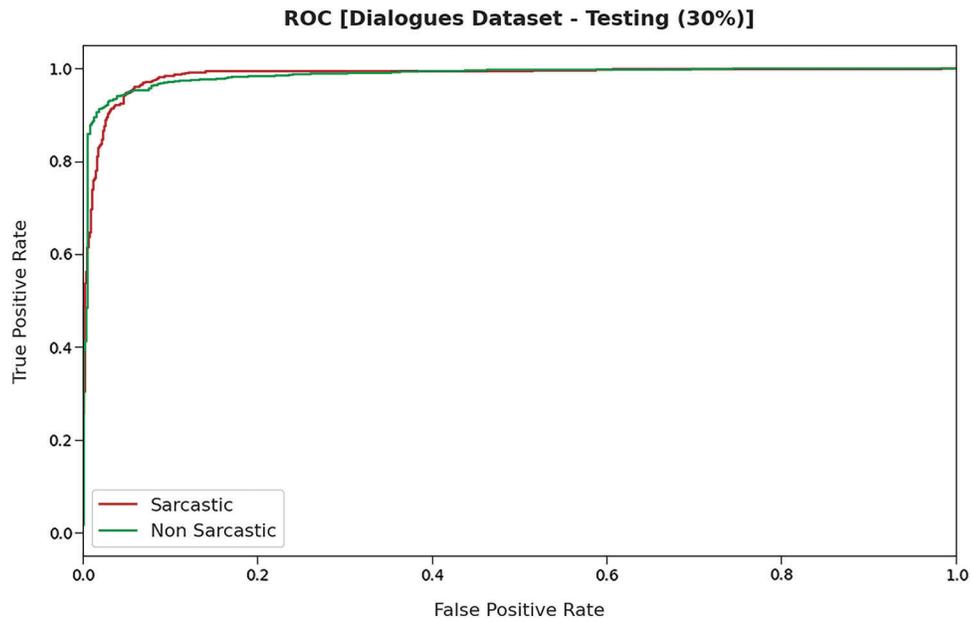
**Figure 10:** Result analysis of DLNLP-SA model on 30% of testing data on Dialogues dataset



**Figure 11:** Precision-recall analysis of DLNLP-SA model on 30% of testing data on Dialogues dataset

A detailed comparison study of the DLNLP-SA model with recent methods on Dialogue dataset is portrayed in [Tab. 4](#) and [Fig. 13](#). The experimental results indicated that the Attention-LSTM and SIARN models have obtained lower classification outcomes over the other methods. Also, the MIARN and ELMo-BiLSTM methods have reached slightly improved performance. Next, the IMHSAA technique has accomplished reasonable outcome. At last, the proposed DLNLP-SA technique demonstrates the enhanced result with  $prec_n$ ,  $reca_l$ ,  $accu_y$ , and  $F_{score}$  of 94.53%, 94.43%, 94.32%, and 94.32% correspondingly.

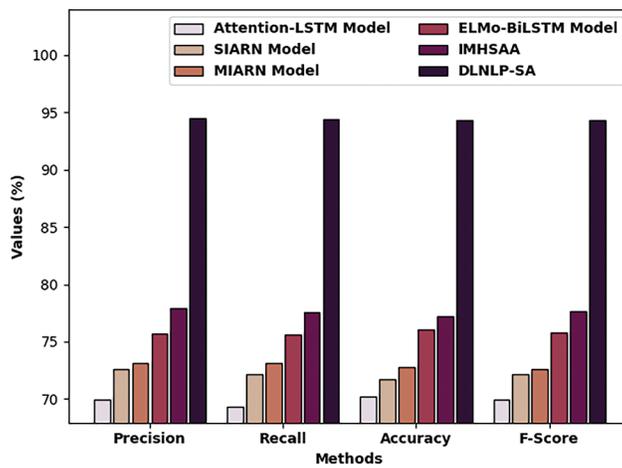
By investigating these results and discussion, it can be concluded that the DLNLP-SA model has accomplished maximum performance on the test Twitter dataset and Dialogues dataset.



**Figure 12:** ROC analysis of DLNLP-SA model on 30% of testing data on Dialogues dataset

**Table 4:** Comparative analysis of DLNLP-SA technique with recent approaches on Dialogues dataset

Methods	Precision	Recall	Accuracy	F-Score
Attention-LSTM Model	69.93	69.31	70.18	69.92
SIARN Model	72.62	72.18	71.74	72.19
MIARN Model	73.10	73.12	72.78	72.62
ELMo-BiLSTM Model	75.68	75.62	76.02	75.82
IMHSAA	77.92	77.57	77.22	77.66
DLNLP-SA	94.53	94.43	94.32	94.32



**Figure 13:** Comparative analysis of DLNLP-SA technique on Dialogues dataset

## 5 Conclusion

In this study, a novel DLNLP-SA technique aims for detecting and classifying the occurrence of sarcasm in the input data. The proposed DLNLP-SA technique undergoes distinct set of processes such as pre-processing, N-gram feature extraction, MHSA-GRU based classification, and MFO based hyperparameter optimization. The application of the MFO algorithm aids in the effectual selection of hyperparameters involved in the MHSA-GRU model. For investigating the improved performance of the DLNLP-SA model, a comprehensive set of simulations were executed using benchmark dataset and the outcomes signified the supremacy over the existing approaches. Therefore, the DLNLP-SA model has been utilized as a proficient tool for SD and classification. In future, the detection efficiency can be improved by hybrid DL models.

**Acknowledgement:** This work was supported through the Annual Funding track by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Project No. AN000685].

**Funding Statement:** This work was supported through the Annual Funding track by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Project No. AN000685].

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] D. Vinoth and P. Prabhavathy, "An intelligent machine learning-based sarcasm detection and classification model on social networks," *The Journal of Supercomputing*, vol. 19, no. 1–2, pp. 288, 2022.
- [2] L. K. Ahire, S. D. Babar and G. R. Shinde, Sarcasm detection in online social network: Myths, realities, and issues. In: *Security Issues and Privacy Threats in Smart Ubiquitous Computing, Studies in Systems, Decision and Control Book Series*. Vol. 341. Singapore: Springer, pp. 227–238, 2021.
- [3] L. H. Son, A. Kumar, S. R. Sangwan, A. Arora, A. Nayyar *et al.*, "Sarcasm detection using soft attention-based bidirectional long short-term memory model with convolution network," *IEEE Access*, vol. 7, pp. 23319–23328, 2019.
- [4] D. Jain, A. Kumar and G. Garg, "Sarcasm detection in mash-up language using soft-attention based bi-directional LSTM and feature-rich CNN," *Applied Soft Computing*, vol. 91, no. 1–2, pp. 106198, 2020.
- [5] N. Pawar and S. Bhingarkar, "Machine learning based sarcasm detection on twitter data," in *2020 5th Int. Conf. on Communication and Electronics Systems (ICCES)*, Coimbatore, India, pp. 957–961, 2020.
- [6] A. Kumar, V. T. Narapareddy, V. A. Srikanth, A. Malapati and L. B. M. Neti, "Sarcasm detection using multi-head attention based bidirectional lstm," *IEEE Access*, vol. 8, pp. 6388–6397, 2020.
- [7] R. A. Potamias, G. Siolas and A. G. Stafylopatis, "A transformer-based approach to irony and sarcasm detection," *Neural Computing and Applications*, vol. 32, no. 23, pp. 17309–17320, 2020.
- [8] S. M. Sarsam, H. A. Samarraie, A. I. Alzahrani and B. Wright, "Sarcasm detection using machine learning algorithms in Twitter: A systematic review," *International Journal of Market Research*, vol. 62, no. 5, pp. 578–598, 2020.
- [9] Y. Du, T. Li, M. S. Pathan, H. K. Teklehaimanot and Z. Yang, "An effective sarcasm detection approach based on sentimental context and individual expression habits," *Cognitive Computation*, vol. 14, no. 1, pp. 78–90, 2022.
- [10] Y. Zhang, Y. Liu, Q. Li, P. Tiwari, B. Wang *et al.*, "CFN: A complex-valued fuzzy network for sarcasm detection in conversations," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 12, pp. 3696–3710, 2021.
- [11] Z. Wen, L. Gui, Q. Wang, M. Guo, X. Yu *et al.*, "Sememe knowledge and auxiliary information enhanced approach for sarcasm detection," *Information Processing & Management*, vol. 59, no. 3, pp. 102883, 2022.

- [12] M. Bedi, S. Kumar, M. S. Akhtar and T. Chakraborty, "Multi-modal sarcasm detection and humor classification in code-mixed conversations," *IEEE Transactions on Affective Computing*, pp. 1, 2021. <http://dx.doi.org/10.1109/TAFFC.2021.3083522>.
- [13] L. Ren, B. Xu, H. Lin, X. Liu and L. Yang, "Sarcasm detection with sentiment semantics enhanced multi-level memory network," *Neurocomputing*, vol. 401, no. 1–2, pp. 320–326, 2020.
- [14] Y. Zhang, Y. Liu, Q. Li, P. Tiwari, B. Wang *et al.*, "CFN: A complex-valued fuzzy network for sarcasm detection in conversations," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 12, pp. 3696–3710, 2021.
- [15] D. K. Nayak and B. K. Bolla, Efficient deep learning methods for sarcasm detection of news headlines. In: *Machine Learning and Autonomous Systems, Smart Innovation, Systems and Technologies book series*. Vol. 269. Singapore: Springer, pp. 371–382, 2022.
- [16] V. Govindan and V. Balakrishnan, "A machine learning approach in analysing the effect of hyperboles using negative sentiment tweets for sarcasm detection," *Journal of King Saud University-Computer and Information Sciences*, vol. 124, no. 1, pp. 109781, 2022.
- [17] R. Zhao, D. Wang, R. Yan, K. Mao, F. Shen *et al.*, "Machine health monitoring using local feature-based gated recurrent unit networks," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1539–1548, 2018.
- [18] R. Akula and I. Garibay, "Interpretable multi-head self-attention architecture for sarcasm detection in social media," *Entropy*, vol. 23, no. 4, pp. 394, 2021.
- [19] K. Zervoudakis and S. Tsafarakis, "A mayfly optimization algorithm," *Computers & Industrial Engineering*, vol. 145, no. 5, pp. 106559, 2020.
- [20] E. Riloff, A. Qadir, P. Surve, L. D. Silva, N. Gilbert *et al.*, "Sarcasm as contrast between a positive sentiment and negative situation," in *Proc. of the 2013 Conf. on Empirical Methods in Natural Language Processing*, Seattle, Washington, USA, pp. 704–714, 2013.
- [21] S. Oraby, V. Harrison, L. Reed, E. Hernandez, E. Riloff *et al.*, "Creating and characterizing a diverse corpus of sarcasm in dialogue," in *Proc. of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Los Angeles, CA, USA, pp. 31–41, 2016.