IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 31 July 2023, accepted 11 September 2023, date of publication 14 September 2023, date of current version 28 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3315609

RESEARCH ARTICLE

Moth Flame Optimization With Hybrid Deep Learning Based Sentiment Classification Toward ChatGPT on Twitter

MOHAMMED ALJEBREEN[1], BAYAN ALABDUALLAH[2], MASHAEL M. ASIRI[3], AHMED S. SALAMA[4], MOHAMMED ASSIRI[5], AND SARA SAADELDEEN IBRAHIM[6]

[1]Department of Computer Science, Community College, King Saud University, P.O. Box 28095, Riyadh 11437, Saudi Arabia
[2]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[3]Department of Computer Science, College of Science and Art at Mahayil, King Khalid University, Abha 61421, Saudi Arabia
[4]Department of Electrical Engineering, Faculty of Engineering and Technology, Future University in Egypt, New Cairo 11845, Egypt
[5]Department of Computer Science, College of Sciences and Humanities-Aflaj, Prince Sattam bin Abdulaziz University, Al-Aflaj, Al-Kharj 16273, Saudi Arabia
[6]Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia

Corresponding author: Mohammed Assiri (m.assiri@psau.edu.sa)

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through large group Research Project under grant number (RGP2/65/44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R440), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Research Supporting Project number (RSP2023R459), King Saud University, Riyadh, Saudi Arabia. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444). This study is partially funded by the Future University in Egypt (FUE).

ABSTRACT ChatGPT, developed by OpenAI, is an advanced language model that excels at generating human-like text responses in conversational settings. As ChatGPT interacts with the user, it creates a range of sentiments from them, involving neutral, positive, or negative expressions. Sentiment analysis (SA), also called opinion mining, is a branch of natural language processing (NLP) that defines the emotional tone or sentiment conveyed in textual data. Sentiment analysis (SA) plays a major role in understanding how people respond and perceive different entities, involving services, products, brands, or artificial intelligence (AI) models GPT. Analyzing the sentiment toward ChatGPT gives valuable insight into, user experience, areas, and satisfaction for development. The study presents a moth flame optimization with hybrid deep learning-based sentiment analysis (MFOHDL-SA) on ChatGPT. The major aim of the MFOHDL-SA method is to design an automated AI model to properly classify the tweets as positive, negative, or neutral in sentiment towards ChatGPT. To accomplish this, the MFOHDL-SA technique initially pre-processes the tweets in different stages. Next, the TF-IDF model is used for the word embedding process. Moreover, the HDL method comprising a convolutional neural network with long short-term memory (CNN-LSTM) method was utilized for sentiment classification. To improve the classifier results of the HDL model, the MFO algorithm is used for hyperparameter tuning. The simulation results of the MFOHDL-SA technique are validated on the Twitter dataset from the Kaggle repository. The obtained experimental outcomes stated the betterment of the MFOHDL-SA approach over other existing techniques in terms of different measures. This provides a valued understanding of public sentiment towards ChatGPT on Twitter, allowing improved understanding and assessment of its impact and perception among users.

INDEX TERMS Artificial intelligence, ChatGPT, sentiment analysis, deep learning, machine learning, data classification.

I. INTRODUCTION

As a field of science, Artificial intelligence (AI) focused on developing systems that act and think like humans, is the driving force behind communication and information technologies in Industry 4.0. With the escalating use of AI technologies, they were used in different fields namely cyber security, industry, education, defence, military, and healthcare [1]. Moreover, AI can be utilized by all sections of society for purposes like entertainment, education, and sports.

The associate editor coordinating the review of this manuscript and approving it for publication was Giacomo Fiumara.

The pervasive use of mobile internet has enabled users to apply AI technologies like voice recognition and image processing in day-to-day life by using tablets and smartphones, mainly in social media platforms and banking [2]. Furthermore, users can control smart home devices i.e. robot vacuums, televisions, air conditioning systems, and boilers, through the Internet both outside and inside of the home. The usage of AI in everyday life is not limited to these instances [3]. There is AI-based technology developed for research purposes like analysis studies, translation from foreign languages, plagiarism checking, and information access. Such technologies include the aspects of modelling, natural language processing (NLP), pattern recognition, robotics, and machine learning (ML) [4]. This study focuses on a conversational AI chatbot known as ChatGPT and NLP approaches.

A large language model refers to a form of generative AI, which produces human-like language [5]. OpenAI trains its text-generating methods by making use of ML approaches on huge quantities of text, websites, books, articles, Wikipedia, and news. Methods get the structures and pattern of language by processing terabytes of data, letting them offer users meaningful and appropriate content as per their requests [6]. Whenever one asks ChatGPT to define itself (prompt-Describe ChatGPT), it will generate the italicized text: ChatGPT can be a variant of the Generative Pre-training Transformer method (GPT), a language model developed by OpenAI and that can be trained on an enormous quantity of textual dataset for generating human-like text [7]. It is utilized for numerous NLP tasks like question answering, language translation, chatbot applications, and text summarization. It can respond and understand to human input informally, making it ideally suitable for interactive applications. It is fine-tuned on particular tasks and datasets that are personalized to execute particular tasks and generate outputs that are highly useful and precise [8]. ChatGPT goes viral on social networking sites, and anecdotal evidence recommends that students are familiar with it. It is considered safe that undergraduate students were utilizing a similar software or ChatGPT. Currently, there is tremendous growth in the use of microblogging platforms, one of them being Twitter [9]. Because of this progression, media outlets and businesses are looking for approaches to use Twitter to collect data on how people perceive their services or products [10]. Though there was research on how sentiments were communicated in genres like online reviews and news articles, there was far less research on how sentiments were expressed in informal language and microblogging because of message length limits.

This study presents a moth flame optimization with hybrid deep learning-based sentiment analysis (MFOHDL-SA) on ChatGPT. The MFOHDL-SA technique aims to design an automated AI model to properly classify the tweets as positive, negative, or neutral in sentiment towards ChatGPT. To accomplish this, the MFOHDL-SA technique initially preprocesses the tweets in different stages. Next, the TF-IDF model is used for the word embedding process. Moreover, the HDL method comprising convolutional neural network with long short-term memory (CNN-LSTM) method is utilized for sentiment classification. To improve the classifier results of the HDL model, the MFO algorithm is used for hyperparameter tuning. The simulation results of the MFOHDL-SA technique are validated on the Twitter dataset from the Kaggle repository

II. RELATED WORKS

In [11], utilized DL approaches to apprehend the insight of scholars in the medical field regarding the newly developed chat generative pre-trained transformer (ChatGPT). With the pretrained BERT i.e. Bidirectional Encoder Representations from Transformers method, the author accomplished SA and topic modelling for examining social media posts of medical researchers for understanding their emotions towards ChatGPT. Feng et al. [12] devise a crowdsource, a data-driven structure that uses 2 social media sites, Reddit and Twitter, for exploring the effect of ChatGPT. With the wide study of social media data gathered from Reddit and Twitter, the author revealed how ChatGPT is converting streaming media from different viewpoints.

Lee et al. [13] intend at identifying Tweets that have racist texts by executing the SA of Tweets. Due to the better DL performance, a stacked ensemble DL method was assembled by merging GRU, RNN, and CNN named, Gated Convolutional Recurrent- Neural Networks (GCR-NN). For extracting the prominent and suitable features from raw text, GRU is on the top in the GCR-NN method, and CNN extracted significant features for RNN for precise estimations. In [14], introduces an optimization-related ML method for categorizing Twitter data. The procedure was accomplished in three phases. During the initial stage, data is preprocessed and collected, in the next stage, the data was enhanced by deriving essential features, and during the last stage the updated training sets are categorized into various classes by implementing various ML approaches.

Biradar et al. [15] developed big data technology utilized to gather and handle large unstructured datasets from real-time social media for SA to ascertain the services and brand. Based on SA, the method devised with the use of customer review classification dealt with prepossessing data, clustering of data depending on particular fields, feature vector leveraging n-gram methods, and TF-IDF vector extract synonyms and classification SA. In [16], designed the usage of historical in addition to sentiment datasets to proficiently forecast stock prices by implementing LSTM.

Parimala et al. [17] perform SA on the tweets in a particular disaster context for specific places at various intervals of time. Depending on the context and the history of tweets, the LSTM network with a word embedding approach was leveraged for deriving keywords. The presented algorithm risk assessment SA (RASA) make use of the keywords from the network for categorizing the sentiment score and tweets for all location detected. In [18], identifies all tweet words and the author assigns a meaning to it. The featured work was integrated with stop words, tweet words, and word2vec, and compiled into the DL algorithms of the CNN method and LSTM, these methods can find the pattern of stop word count with their method.

Devarajan et al. [19] presented a novel AI-assisted fake news recognition with deep NLP approach. The presented work is considered in 4 layers such as enabled edge layer, publisher layer, cloud layer, and social media networking layer. In [20], the author's concentration is about extraction feature of our data with hybrid method of integrating LR with dimensional reduction approach employing PCA. In [21], the authors gathered 600 million public tweets utilizing URL-based security device and feature generation has been executed for SA. The ternary classification was processed dependent upon pre-processing method, and the outcome of tweets sent by users are attained. The authors employ a hybridization system employing 2 optimizer methods and one ML approaches like PSO, GA, and DT for classification accuracy by SA. In [22], a novel hybrid system was presented for detecting the stream of Twitter spam in real-time employing the integration of DT, PSO, and GA. Twitter has provide access to researchers to get tweets in its Twitter-API for realtime streaming of tweet data that it can obtain direct access to open tweets.

III. THE PROPOSED MODEL

In this study, we have developed the MFOHDL-SA method for sentiment classification on ChatGPT. The goal of the MFOHDL-SA technique is to design an automated AI model to properly 0classify the tweets as positive, negative, or neutral in sentiment towards ChatGPT. Fig. 1 represents the overall procedure of the MFOHDL-SA system. The figure states that the MFOHDL-SA technique comprises different stages of operations namely preprocessing, TF-IDF model, HDL-based sentiment classification, and MFO-based hyperparameter tuning. Initially, the input data gets preprocessed to improve the quality of the data. Next, the TF-IDF model is applied for word embedding process. For sentiment classification, the HDL model is used and its hyperparameters can be chosen by the use of MFO algorithm.

A. DATA PRE-PROCESSING

Text preprocessing is a vital process of Natural Language Processing (NLP), which recreates text into a digestible form for ML algorithm [23]. The text preprocessing step includes converting emojis into words, removal of stop words, removing punctuations, retweets, and URLs, removing collection words, lemmatization, stemming, and tokenization.

1) RETWEETS

A retweet is when a tweeter user shared a tweet from other users. Sharing of tweets leads to duplication. Duplicate tweets may rise space requirements to perform a scientific procedure, impede the model fitness, and skew the word frequency. Thus, retweets were removed as they have a content duplication.

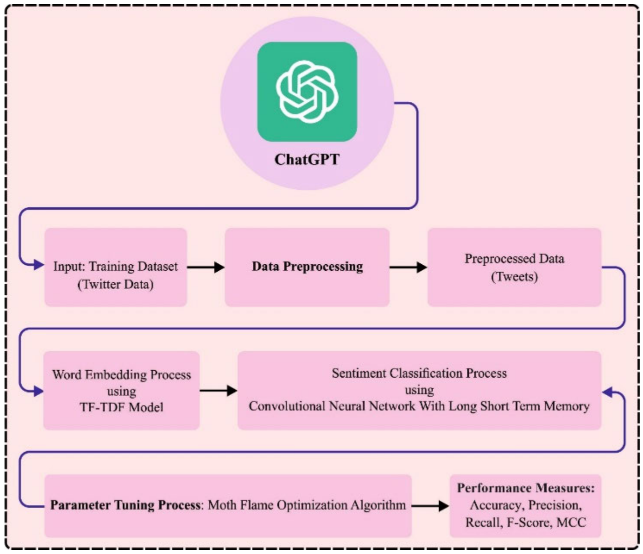

FIGURE 1. Overall process of the MFOHDL-SA approach.

2) URLS AND PUNCTUATION

Removing URLs is the next step. Here, URLs were removed from the tweet. Cleaning URLs might be crucial because it doesn't have meaning and doesn't affect sentimental values. Keeping the URL link might skew the word frequency since every single tweet has the link. Cleaning up tweet text involves the removal of non-useful punctuations and signs, including #,?/, \,! using Python re-package.

3) EMOJI

The next step is to convert emoji into words utilizing emoji. demojize() from the Python library and afterwards clean up tweets from the retweets. Emojis are often used to show the emotions of people. Thus, these emoji are converted into phrases that might enhance the SA of the tweets.

4) TOKENIZATION AND NORMALIZATION

Tokenization is used to split the words in every single tweet. Here, combining the text tokenization with text normalization for reducing the length of the code. Each set of words from tweets is transformed into lowercase as normalized words. Then, store the words from the tweets for computing the polarity.

5) STOP WORDS

Afterwards splitting words from the tweet, the next step was eliminating stop words. A sklearn package, 'stopwords', was utilized for removing stop words in English. Removal of stop words, including 'is', 'am', aren't, hasn't, and so on., are useful to measure the sentiment of the tweets because the stop words aren't convenient for analysis.

6) STEMMING AND LEMMATIZATION

The stemming process is used to reduce the words into the base, root form, or word stem. For instance, words including looks, looking, or looked were reduced to look. At the same time, this process reduces inflection in words into the root form. Sometimes, words might be invalid in the language. Hence, the study used the 'PorterStemmer' stemming process. Different from stemming, the lemmatization process is used to reduce inflected words, which ensures that the root word belongs to the language. Lemmatization is the form of words and is mainly based on vocabulary.

B. FEATURE EXTRACTION

Once the tweets are pre-processed, the next stage is to generate word vectors using the TF-IDF model. TFIDF comprises two parts, IDF and TF, where TF identifies word frequency, and IDF shows the frequency of word that is accessible in the text documents [24]. If a word, like 'is', 'is', or 'am', takes place in different texts, then IDF values are lower. At the same time, if the word arises in a smaller amount of text, then the IDF value is lower. In the meantime, IDF is extremely leveraged to define the word importance. Consider TF as the word frequency and it was shown below

$$TP = \frac{N(d)}{N} \tag{1}$$

In Eq. (1), $N(d)$ specifies the count of entries d in all the classes and N denotes the total amount of entries.

The IDF denotes inverse text frequency and it is calculated as follows,

$$IDF\ (d) = \log\frac{N+1}{N\ (d)+1} + 1 \tag{2}$$

In Eq. (2), (d) denotes the overall amount of text that has d word in the repository and N shows the overall amount of texts available in the corpus N.

$$F_3 = TF - IDF\ (d) = TP \times IDF\ (d) \tag{3}$$

C. SENTIMENT CLASSIFICATION

In this work, the classification of sentiments towards ChatGPT takes place using the HDL model. CNN is a new DL model that is popular for its exceptional capability in the region of pattern detection [25]. The convolution layer considers a filter or a kernel and runs it over data and generates a feature map,

$$(f * g)_{(c_1,c_2)} = \sum_{c_1,c_2} f(a_1, a_2)\cdot g\,(b_1, b_2) \tag{4}$$

In Eq. (4), f is data, *and* g refers to the kernel, $a_1 + b_1 = c1$ and $a_2 + b_2 = c2$. Thus, complex convolution functions are positioned to simplify the collected feature afterwards the convolutional layer pooling layer. Generally, the FC layer was positioned at the end of the structure and has the accountability of classification into corresponding groups.

CNN-LSTM is a hybrid DL method which combines the power of a Convolution network and LSTM. The feature extracted by CNN is fed to LSTM, an RNN architecture for capturing the temporal dependency. Fig. 2 depicts the infrastructure of CNN-LSTM.

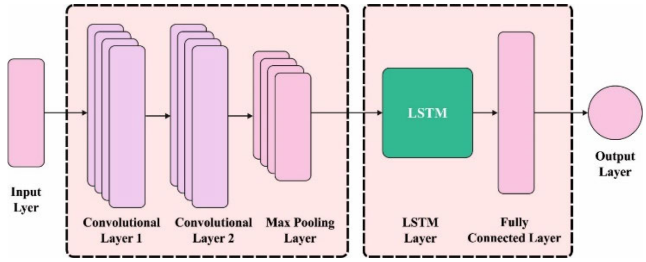

FIGURE 2. Structure of CNN-LSTM.

Memory cell C_t is the critical remodelling in RNN. By using the following equations, updating of LSTM unit working can be governed:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{5}$$
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{6}$$
$$C_t emp = \tanh\left(W_C \cdot [h_{t-1}, x_t] + b_C\right) \tag{7}$$
$$c_t = f_t * c_{t-1} + i_t * c_t emp \tag{8}$$
$$O_t = \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \tag{9}$$
$$h\_t = o\_t * tanh\,(C\_t) \tag{10}$$

where W represent the weighted matrix of the diverse gate with subscript representing the particular gate i, f, O, and C representing the input gate, forget gate, output gate, and cell activation vectors correspondingly, having the similar size as hidden state vector h. W_C denotes the weight of the activation cell.

For compensating for the shortcomings of typical cross-entropy loss because of data imbalance, a modulating factor can be multiplied to cross-entropy loss for obtaining focal loss.

$$FL\ (p_t) = -(1-p_t)^{\gamma} log\ (p_t)$$

where tunable focusing parameter $\gamma \geq 0$.

IV. HYPERPARAMETER TUNING

Finally, the MFO technique is used for the hyperparameter tuning of the HDL method. MFO is a population-based metaheuristic approach stimulated by moth behaviours, primarily their sloping movements [26]. The MFO technique behaves like the moths and flames, where the moth denotes the solution and the moth's position in the search space indicates the problem variable. Here, the moth specifies search mediators that attempt to identify a schedule in the search space. The moth group searches different places within the search region by notifying the position. Meanwhile, the frames indicate the best location of individual moths. The MFO technique applies four arrays to simulate the moths and fames in the following:

An array with 2D, named M, aims at saving the solution. The number of moths that can be the initial dimension of the array was represented by n and the number of variables in the problem that was the second dimension of this array can be represented by d.

An array with 1D, named OM is intended to maintain the fitness value of the entire individual moth.

An array with 2D, named F, aims at saving the fames that are similar to the M array.

An array with 1D, named OF, aims at preserving the matching fitness rate for all the best positions.

The MFO method has 3 fundamental characteristics: initialization (I), search process (P), and termination (T). Initially, the solution was randomly generated as a population by Eq. (11), where a random trial of the job was produced by the random series of jobs.

$$M\left(i\right) = randomSequenceofJobs\left(0,d-1\right). \quad (11)$$

where OM indicates the moth objective function that equals fitness function (OM = fitness function (M)). d denotes the number of jobs, next, the makespan is evaluated from the fitness function of the moth.

In the local search (or the search procedure) stage, it iteratively keeps searching for the moth solution until the stopping criterion (T) or predefined termination is met. At the termination stage, the solution with minimal value was returned. But the position of the moth was upgraded by using $M_i = S(M_i, F_j)$, where M_i shows the moth, i and j denote indices, S indicates the spiral process, and F_j represents fame.

$$S\left(M_i, F_j\right) = D_i * e^{bt} * cos\left(2\pi t\right) + F_j. \quad (12)$$

where b shows the constant form of a logarithmic spiral, Di characterized the distance from the moth to fame, $Di = \left|F_j - M_i\right|$, and t indicates the randomly generated value within r and 1. The r value linearly declined from −1 to −2.

The MFO was intended for continuous problems. But it extended to the problem of discrete optimization by suggesting moving locations of jobs. The term S stimulates the moth to move nearby the fame. The term S is the jobs that might move. Its value might surpass the problem boundaries in certain conditions. In this case, the move can be reflected toward the random job. On the other hand, the Flame_no (fame number) was evaluated by the following equation:

$$Flame\_no = round\left(N - l * \frac{N-1}{T}\right). \quad (13)$$

where T denotes the number of iterations, l represent the existing number of iteration, and N shows the maximum fame number .

The MFO approach developed a fitness function (FF) that realizes greater classifier outcomes. It solves a positive integer that indicates the good solution of candidate performances. Here, the minimized classifier rate of errors is FF is provided in Eq. (14).

$$\begin{aligned} fitness\left(x_i\right) &= ClassifierErrorRate\left(x_i\right) \\ &= \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \quad (14) \end{aligned}$$

TABLE 1. Details of database.

Class	No. of Samples
Good	20000
Bad	20000
Neutral	20000
Total Samples	60000

TABLE 2. Sentiment classifier outcome of MFOHDL-SA method on 70:30 of TRP/TSP.

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	AUC_{score}	MCC
Training Phase (70%)						
Good	96.87	94.30	96.49	95.38	96.78	93.03
Bad	94.66	91.35	92.73	92.04	94.18	88.03
Neutral	93.73	92.18	88.64	90.37	92.45	85.76
Average	95.09	92.61	92.62	92.60	94.47	88.94
Testing Phase (30%)						
Good	96.85	94.16	96.43	95.28	96.74	92.93
Bad	94.67	91.78	92.34	92.06	94.09	88.05
Neutral	93.73	91.92	89.18	90.53	92.61	85.87
Average	95.09	92.62	92.65	92.62	94.48	88.95

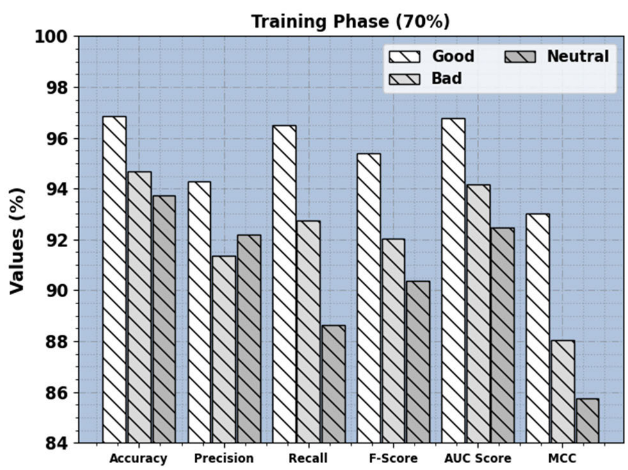

FIGURE 3. Sentiment classifier outcome of MFOHDL-SA approach on 70% of TRP.

V. RESULTS AND DISCUSSION

The proposed model is simulated using Python 3.6.5 tool on PC i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, 250GB SSD, and 1TB HDD. The parameter settings are given as follows: learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50, and activation: ReLU. In this section, the experimental analysis of the MFOHDL-SA method is tested on the sentiment dataset from the Kaggle repository [27]. The dataset holds 60000 samples with three classes as defined in Table 1.

Table 2 represents a detailed sentiment classification result of the MFOHDL-SA technique. Fig. 3 exhibits the classification outcomes of the MFOHDL-SA methodology under

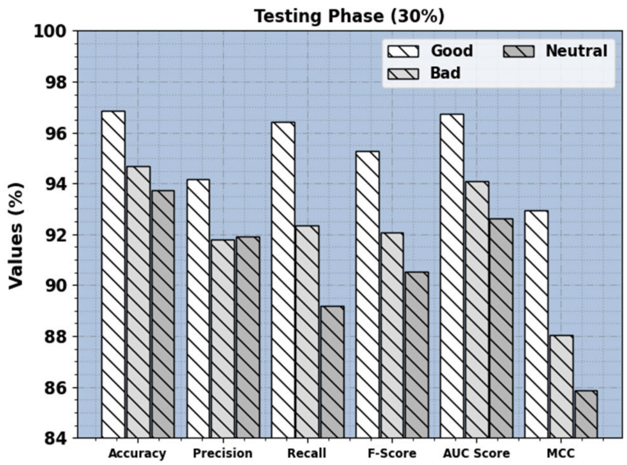

FIGURE 4. Sentiment classifier outcome of MFOHDL-SA approach on 30% of TSP.

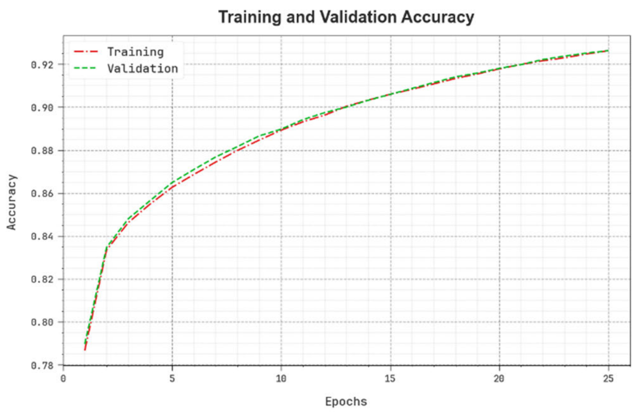

FIGURE 5. Accuracy curve of the MFOHDL-SA method.

70% of TRP. The results inferred that the MFOHDL-SA technique recognizes three types of sentiments proficiently. On good class, the MFOHDL-SA technique attains $accu_y$ of 96.87%, $prec_n$ of 94.30%, $reca_l$ of 96.49%, F_{score} of 95.38%, AUC_{score} of 96.78%, and MCC of 93.03%. Also, on bad class, the MFOHDL-SA method reaches $accu_y$ of 94.66%, $prec_n$ of 91.35%, $reca_l$ of 92.73%, F_{score} of 92.04%, AUC_{score} of 94.18%, and MCC of 88.03%. At last, on neutral class, the MFOHDL-SA approach achieves $accu_y$ of 93.73%, $prec_n$ of 92.18%, $reca_l$ of 88.64%, F_{score} of 90.37%, AUC_{score} of 92.45%, and MCC of 85.76%.

Fig. 4 exhibits the classification outcomes of the MFOHDL-SA approach under 30% of TSP. The outcomes specify the MFOHDL-SA method recognizes three types of sentiments proficiently. On good class, the MFOHDL-SA algorithm reaches $accu_y$ of 96.85%, $prec_n$ of 94.16%, $reca_l$ of 96.43%, F_{score} of 95.28%, AUC_{score} of 96.74%, and MCC of 92.93%. Similarly, on bad class, the MFOHDL-SA method achieves $accu_y$ of 94.67%, $prec_n$ of 91.78%, $reca_l$ of 92.34%, F_{score} of 92.06%, AUC_{score} of 94.09%, and MCC of 88.05%. At last, on neutral class, the MFOHDL-SA approach achieves $accu_y$ of 93.73%, $prec_n$ of 91.92%, $reca_l$ of 89.18%, F_{score} of 90.53%, AUC_{score} of 92.61%, and MCC of 85.87%.

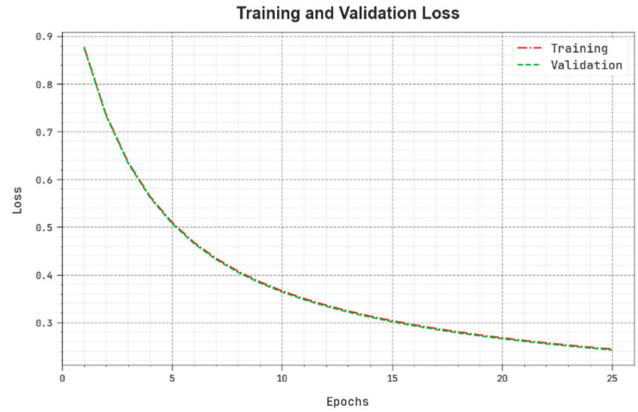

FIGURE 6. Loss curve of the MFOHDL-SA approach.

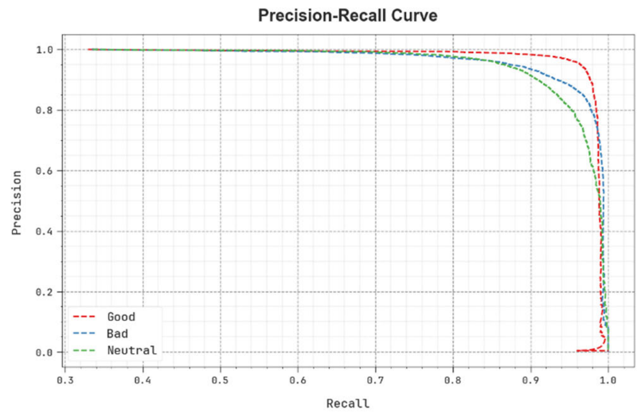

FIGURE 7. PR curve of the MFOHDL-SA approach.

Fig. 5 inspects the $accu_y$ of the MFOHDL-SA algorithm in the training and validation on the test dataset. The result shows that the MFOHDL-SA method attains higher $accu_y$ values over greater epochs. Also, the higher validation $accu_y$ over training $accu_y$ portrays that the MFOHDL-SA algorithm learns productively on the test dataset.

The loss analysis of the MFOHDL-SA approach in the training and validation is presented on the test dataset in Fig. 6. The result demonstrates that the MFOHDL-SA approach acquires closer values of training and validation loss. The MFOHDL-SA system learns productively on the test dataset.

A brief PR curve of the MFOHDL-SA method is validated on the test database in Fig. 7. The result stated the MFOHDL-SA algorithm has higher values of PR. Moreover, the MFOHDL-SA approach can reach greater PR values on every class label.

In Fig. 8, a ROC study of the MFOHDL-SA method is exposed on the test database. The figure described that the MFOHDL-SA method resulted in optimized ROC values. Also, the MFOHDL-SA approach showed the highest ROC values on every class label.

In Table 3 and Fig. 9, the comparative study of the MFOHDL-SA method is clearly stated. The results indicate the supremacy of the MFOHDL-SA technique. Based on

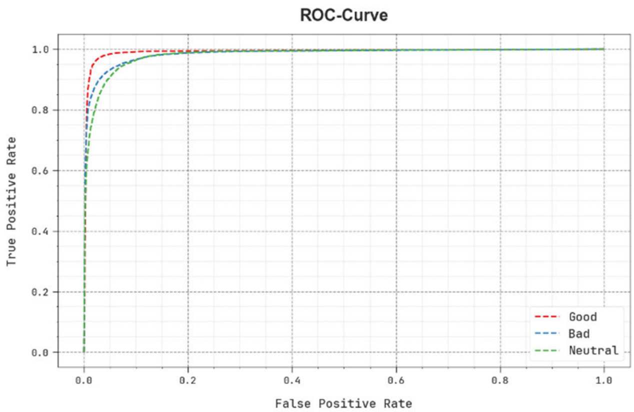

FIGURE 8. ROC curve of the MFOHDL-SA approach.

TABLE 3. Comparative outcome of MFOHDL-SA approach with other methods.

Methodology	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	AUC_{score}
Random Forest	89.70	87.63	87.37	86.57	84.75
Decision Tree	85.38	84.64	84.44	84.75	87.39
SVM Model	90.07	88.89	86.67	87.21	85.85
XGBoost	89.73	89.81	89.74	85.00	86.73
CNN Model	90.91	87.13	89.78	89.91	85.54
ELM Model	86.77	90.86	88.48	84.42	90.57
MFOHDL-SA	95.09	92.62	92.65	92.62	94.48

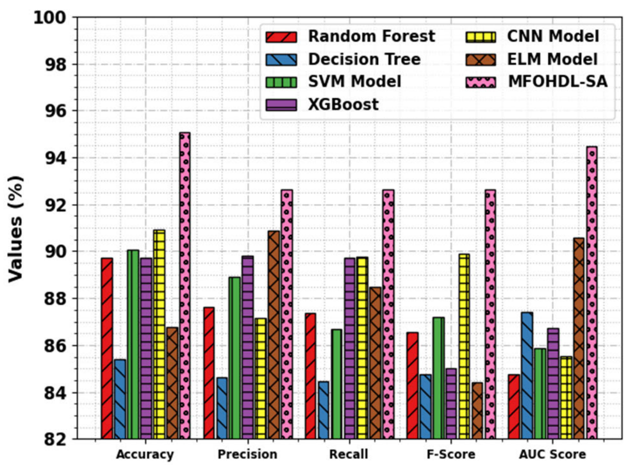

FIGURE 9. Comparative outcome of the MFOHDL-SA method with other approaches.

$accu_y$, the MFOHDL-SA technique gains increasing $accu_y$ of 95.09% while the DT, SVM, XGBoost, CNN, and ELM models obtain decreasing $accu_y$ of 89.70%, 85.38%, 90.07%, 89.73%, 90.91%, and 86.77% respectively.

Meanwhile, based on $prec_n$, the MFOHDL-SA method gains increasing $prec_n$ of 92.62% while the DT, SVM, XGBoost, CNN, and ELM approaches gain decreasing $prec_n$ of 87.63%, 84.64%, 88.89%, 89.81%, 87.13%, and 90.86% correspondingly. Eventually, based on $reca_l$, the MFOHDL-SA method gains increasing $reca_l$ of 92.65% while the DT, SVM, XGBoost, CNN, and ELM approach obtain decreasing $reca_l$ of 87.37%, 84.44%, 86.67%, 89.74%, 89.78%, and 88.48% correspondingly. These outcomes stated the enhanced performance of the MFOHDL-SA approach on sentiment classification towards ChatGPT.

VI. CONCLUSION

In this study, we have developed the MFOHDL-SA method for sentiment classification on ChatGPT. The goal of the MFOHDL-SA technique is to design an automated AI model to properly classify the tweets as positive, negative, or neutral in sentiment towards ChatGPT. To accomplish this, the MFOHDL-SA technique comprises four stages of operations namely preprocessing, TF-IDF model, HDL-based sentiment classification, and MFO-based hyperparameter tuning. To improve the classifier outcomes of the HDL method, the MFO algorithm is used for hyperparameter tuning. The simulation results of the MFOHDL-SA technique are validated on the Twitter dataset from the Kaggle repository. The obtained experimental outcomes stated the improved performance of the MFOHDL-SA algorithm over other existing techniques in terms of different measures. This provides a valued understanding of public sentiment towards ChatGPT on Twitter, allowing improved understanding and assessment of its impact and perception among use. In future, the proposed model can be extended to the classification of COVID-19 related sentiments. In addition, ensemble learning approaches can be used to improve the performance of the proposed model.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through large group Research Project under grant number (RGP2/65/44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R440), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Research Supporting Project number (RSP2023R459), King Saud University, Riyadh, Saudi Arabia. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444). This study is partially funded by the Future University in Egypt (FUE).

REFERENCES

[1] S. V. Praveen and V. Vajrobol, "Can ChatGPT be trusted for consulting? Uncovering Doctor's perceptions using deep learning techniques," *Ann. Biomed. Eng.*, pp. 1–4, May 2023.

[2] A. Haleem, M. Javaid, and R. P. Singh, "An era of ChatGPT as a significant futuristic support tool: A study on features, abilities, and challenges," *BenchCouncil Trans. Benchmarks, Standards Evaluations*, vol. 2, no. 4, Oct. 2022, Art. no. 100089.

[3] A. Korkmaz, C. Aktürk, and T. Talan, "Analyzing the user's sentiments of ChatGPT using Twitter data," *Iraqi J. for Comput. Sci. Math.*, vol. 4, no. 2, pp. 202–214, May 2023.

[4] M. Karanouh, "Mapping ChatGPT in mainstream media: Early quantitative insights through sentiment analysis and word frequency analysis," 2023, *arXiv:2305.18340*.

[5] M. U. Haque, I. Dharmadasa, Z. T. Sworna, R. N. Rajapakse, and H. Ahmad, "'I think this is the most disruptive technology': Exploring sentiments of ChatGPT early adopters using Twitter data," 2022, *arXiv:2212.05856*.

[6] P. Törnberg, "ChatGPT-4 outperforms experts and crowd workers in annotating political Twitter messages with zero-shot learning," 2023, *arXiv:2304.06588*.

[7] S. AlZu'bi, A. Mughaid, F. Quiam, and S. Hendawi, "Exploring the capabilities and limitations of ChatGPT and alternative big language models," in *Artificial Intelligence and Applications*, 2022.

[8] F. Rustam, M. Khalid, W. Aslam, V. Rupapara, A. Mehmood, and G. S. Choi, "A performance comparison of supervised machine learning models for COVID-19 tweets sentiment analysis," *PLoS ONE*, vol. 16, no. 2, Feb. 2021, Art. no. e0245909.

[9] A. Kumar and A. Jaiswal, "Systematic literature review of sentiment analysis on Twitter using soft computing techniques," *Concurrency Comput., Pract. Exper.*, vol. 32, no. 1, p. e5107, Jan. 2020.

[10] A. Tlili, B. Shehata, M. A. Adarkwah, A. Bozkurt, D. T. Hickey, R. Huang, and B. Agyemang, "What if the devil is my guardian angel: ChatGPT as a case study of using chatbots in education," *Smart Learn. Environments*, vol. 10, no. 1, p. 15, Feb. 2023.

[11] S. V. Praveen and V. Vajrobol, "Understanding the perceptions of healthcare researchers regarding ChatGPT: A study based on bidirectional encoder representation from transformers (BERT) sentiment analysis and topic modeling," *Ann. Biomed. Eng.*, vol. 51, pp. 1–3, May 2023.

[12] Y. Feng, P. Poralla, S. Dash, K. Li, V. Desai, and M. Qiu, "The impact of ChatGPT on streaming media: A crowdsourced and data-driven analysis using Twitter and Reddit," in *Proc. IEEE IEEE 9th Int. Conf. Big Data Secur. Cloud (BigDataSecurity) Int. Conf. High Perform. Smart Comput., (HPSC) IEEE Intl Conf. Intell. Data Secur. (IDS)*, May 2023, pp. 222–227.

[13] E. Lee, F. Rustam, P. B. Washington, F. E. Barakaz, W. Aljedaani, and I. Ashraf, "Racism detection by analyzing differential opinions through sentiment analysis of tweets using stacked ensemble GCR-NN model," *IEEE Access*, vol. 10, pp. 9717–9728, 2022.

[14] A. Naresh and P. V. Krishna, "An efficient approach for sentiment analysis using machine learning algorithm," *Evol. Intell.*, vol. 14, no. 2, pp. 725–731, Jun. 2021.

[15] S. H. Biradar, J. V. Gorabal, and G. Gupta, "Machine learning tool for exploring sentiment analysis on Twitter data," *Mater. Today, Proc.*, vol. 56, pp. 1927–1934, 2022.

[16] I. Gupta, T. K. Madan, S. Singh, and A. K. Singh, "HiSA-SMFM: Historical and sentiment analysis based stock market forecasting model," 2022, *arXiv:2203.08143*.

[17] M. Parimala, R. M. S. Priya, M. P. K. Reddy, C. L. Chowdhary, R. K. Poluru, and S. Khan, "Spatiotemporal-based sentiment analysis on tweets for risk assessment of event using deep learning approach," *Softw., Pract. Exper.*, vol. 51, no. 3, pp. 550–570, Mar. 2021.

[18] U. D. Gandhi, P. M. Kumar, G. C. Babu, and G. Karthick, "Sentiment analysis on Twitter data by using convolutional neural network (CNN) and long short term memory (LSTM)," *Wireless Pers. Commun.*, pp. 1–10, May 2021.

[19] G. G. Devarajan, S. M. Nagarajan, S. I. Amanullah, S. A. Mary, and A. K. Bashir, "AI-assisted deep NLP-based approach for prediction of fake news from social media users," *IEEE Trans. Computat. Social Syst.*, early access, Mar. 29, 2023, doi: 10.1109/TCSS.2023.3259480.

[20] N. S. Murugan and G. U. Devi, "Feature extraction using LR-PCA hybridization on Twitter data and classification accuracy using machine learning algorithms," *Cluster Comput.*, vol. 22, no. S6, pp. 13965–13974, Nov. 2019.

[21] S. M. Nagarajan and U. D. Gandhi, "Classifying streaming of Twitter data based on sentiment analysis using hybridization," *Neural Comput. Appl.*, vol. 31, no. 5, pp. 1425–1433, May 2019.

[22] N. S. Murugan and G. U. Devi, "Detecting streaming of Twitter spam using hybrid method," *Wireless Pers. Commun.*, vol. 103, no. 2, pp. 1353–1374, Nov. 2018.

[23] M. Qorib, T. Oladunni, M. Denis, E. Ososanya, and P. Cotae, "COVID-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination Twitter dataset," *Exp. Syst. Appl.*, vol. 212, Feb. 2023, Art. no. 118715.

[24] G. Singh and A. Nagpal, "HFCVO-DMN: Henry fuzzy competitive verse optimizer-integrated deep maxout network for incremental text classification," *Computation*, vol. 11, no. 1, p. 13, Jan. 2023.

[25] V. Bijalwan, V. B. Semwal, G. Singh, and T. K. Mandal, "HDL-PSR: Modelling spatio-temporal features using hybrid deep learning approach for post-stroke rehabilitation," *Neural Process. Lett.*, vol. 55, no. 1, pp. 279–298, Feb. 2023.

[26] A. Abuhamdah, "Modified hybrid moth optimization algorithm for PFSS problem," *Social Netw. Comput. Sci.*, vol. 4, no. 3, p. 298, Mar. 2023.

[27] Accessed: Apr. 23, 2023. [Online]. Available: https://www.kaggle.com/datasets/charunisa/chatgpt-sentiment-analysis