



OPEN Sentiment analysis of tweets employing convolutional neural network optimized by enhanced gorilla troops optimization algorithm

Fang Li¹, Jialing Li²✉ & Francis Abza^{3,4}✉

Sentiment analysis has become a difficult and important task in the current world. Because of several features of data, including abbreviations, length of tweet, and spelling error, there should be some other non-conventional methods to achieve the accurate results and overcome the current issue. In other words, because of those issues, conventional approaches cannot perform well and accomplish results with high efficiency. Emotional feelings, such as fear, anxiety, or traumas, often stem from many psychological issues experienced during childhood that can persist throughout life. In addition, people discuss and share their ideas on social media, often unconsciously representing their hidden emotions in the comments. This study is about sentiment analysis of tweets shared by several people. In fact, sentiment analysis can determine whether the shared comments and tweets are positive or negative. The paper introduces the use of a Convolutional Neural Network (CNN), a kind of neural network, optimized by the Enhanced Gorilla Troops Optimization Algorithm (CNN-EGTO). Two datasets provided by the SemEval-2016 are used to evaluate the system, while the polarity of tweets were manually determined. It was determined by the findings of the present study that the suggested model could approximately achieve the values of 98%, 95%, 98%, and 96.47% for accuracy, precision, recall, and F1-score, respectively, for positive polarity. In addition, the suggested model could gain the values of 97, 96, 98, and 97.49 for precision, recall, accuracy, and F1-score, respectively, for negative polarity. Consequently, it was found that the suggested model could outperform the other models by considering their performance and efficiency. These values of performance metrics represent that the suggested model could determine the polarity of sentence, positive or negative, with great efficiency.

Keywords Sentiment analysis, Twitter, Convolutional neural network, Enhanced gorilla optimization algorithm, Supervised learning

The rise and utilization of the Internet has significantly transformed the way individuals express their viewpoints on various subjects. This transformation has been influenced by platforms, such as email and social media. One example is the impactful role of social media as a tool for sharing and communicating information on the Internet. In other words, it helps the exchange of information and the formation of new connections, allowing people to freely share information through writing brief messages on their product review websites, online discussion platforms, and walls¹. To do so, Businesses, governments, and other organizations employ sentiments that have been explained on platforms of social media. For instance, companies have the ability to monitor the effectiveness of their offerings by analyzing feedback received from social media. They have the capability to collect valuable information and business knowledge to enhance the quality of their future services and products. Furthermore, they can distinguish potential customers from the overall audience and conduct segmentation of market to make more informed business decisions^{2,3}. Therefore, it can highlight the influence of sentiment analysis that many organizations can use it to enhance quality of their products, sales, and so on.

¹Global Business School, Chongqing College Of International Business And Economics, Chongqing 401520, China.

²School of Artificial intelligence, Chongqing Youth Vocational & Technical College, Chongqing 401320, China.

³University of Ghana, P.O. Box 134, Legon-Accra, Ghana. ⁴College of Technical Engineering, The Islamic University, Najaf, Iraq. ✉email: candytddy@163.com; francisabzaf@gmail.com

Sentiment analysis is the process of ascertaining opinions, feelings, and viewpoints expressed in source materials⁴, such as short texts, documents, sentences from blogs^{5,6}, news⁷, and reviews^{8–10}, amid other outlets, which are about hot news like Covid19¹¹. This field often involves processing extensive text collections and usually deals with “formal language”. Sentiment analysis is a field that employs NLP (Natural Language Processing) and machine learning to identify the collective sentiment towards a product or topic¹². Typically, sentiment analysis involves using a feature set derived from the original dataset.

Sentiment analysis is also commonly conducted by utilizing a feature set derived from the original dataset¹³. Nevertheless, the efficacy, particularly accuracy, has been frequently impacted by the curse of dimensionality, noise, the size of data, and the data domains employed for testing and training the models¹⁴. This study seeks to create a model that integrates dimensionality decrease, the utilization of various parts of speech, and NLP for analysis of sentiment. Additionally, it should be noted that the dataset from Twitter platform has been employed in this study.

Considering increasing popularity of Twitter, all tweets could have obtained considerable power for sentiment analysis articles¹⁵. In fact, Twitter allows users to share short status updates known as “tweets” of up to 140 characters. Users typically post messages that are well below this character limit. With around 200 million users, Twitter hosts a vast and constantly changing body of user-produced content, with approximately 400 million tweets being posted each day¹⁶. Over the past few years, individuals have frequently shared their experiences and ideas on various social media platforms, comprising Facebook, Twitter, and Instagram. Several studies are carried out with the purpose of employing NLP approaches for processing text data on the basis of internet, like tweets, text messages, and posts relevant to health issues^{17–20}. Although Twitter is a good platform to use, it has several challenges as well.

Working with tweets using NLP approaches presents several main challenges. The main challenge involves the abundance of novel posts online, coupled with the limited availability of APIs. With up to 90 or even 100 million tweets each day²¹, research is often based on stochastic instances¹⁶. Besides, there are three diverse approaches, which are used to do sentiment analysis tasks. In general, there are three primary approaches to sentiment analysis, including supervised approach (Machine Learning-based Method), a hybrid method, and an unsupervised approach (Lexicon-based Method)²².

When discussing sentiment analysis of content of social media, many academic papers utilize a machine learning approach²³. This involves using a labeled dataset for training a conventional optimizer. A significant issue with this technique is the challenge of developing suitable datasets, since the trained categorizers often have strong ties to specific domains²². Furthermore, there have been some studies of scholars discussed in the following to go deep into the topic more.

Literature review

Jianqiang et al.²⁴ suggested a word embedding approach that was gained by unsupervised learning on the basis of massive corpora of twitter. This approach utilized statistical co-occurrence features and latent contextual semantic connections between words within various tweets. The word embedding was merged with word sentiment polarity scores and n-grams attributes to create a set of sentiment attributes for tweets. This attribute set was incorporated into a deep CNN for forecasting and training sentiment categorization labels. Some experimental comparisons were conducted between the model and the baseline models, which used a word n-grams network, on 5 different datasets of Twitter. The outcomes showed that the model outperformed the baseline model in terms of F1-measure and accuracy for Twitter sentiment categorization. Although, the study depended on CNN, which excelled at capturing local patterns, it could not capture long-range context or dependencies. Considering sentiment analysis, comprehending full of a tweet or sentence, like sentiment shifts or negations, is highly essential; however, CNN may not have good performance in this regard.

Umer et al.²⁵ proposed an integration of CNN and Long Short-Term Memory (LSTM) to conduct analysis of sentiment on datasets of Twitter. The proposed model's efficacy was assessed using machine learning categorizers, such as the Random Forest (RF), support vector classifier, logistic regression, Stochastic Gradient Descent (SGD), a Voting Classifier (VC) of SGD and RF, as well as other advanced classifier networks. Additionally, two feature extraction approaches (word2vec and term frequency-inverse document frequency) were examined to ascertain their influence on forecast accuracy. 3 datasets (hate speech, women's e-commerce reviews of clothing, and US airline sentiments) were employed to assess the proposed model's efficiency. Experimental findings indicated that the CNN-LSTM achieved better accuracy compared to other categorizers. On the other hand, it is true that the integration of LSTM and CNN can raise the performance of the model, but the complexity gets raised as well. It should be noted that training this hybrid model is really expensive and time-consuming, which may limit practicality and scalability.

Another study was suggested by Tam²⁶, in which the combination framework of Bi-LSTM and CNN was employed. The implementation of ConvBiLSTM included a word embedding network that transformed tweets into numbers. The CNN layer received the feature embedding as input and generated minor dimensions of feature. The Bi-LSTM model then took the input and generated the categorization outcome. GloVe and Word2Vec were employed separately to assess their effect on the results of word embedding within the suggested network. ConvBiLSTM was used with both SST-2 datasets and retrieved Tweets. When ConvBiLSTM was applied with Word2Vec on the retrieved dataset of Tweets, it performed better than the other networks with an accuracy of 91.13%. This study, too, had several problems in terms of dealing with sentiment analysis task. The study did not mention that how misspellings or abbreviations were dealt with in the stage of preprocessing that poor preprocessing stage can lead to reduced performance.

Gandhi et al.²⁷ identified and assigned a meaning to each tweet. The feature work has been integrated with words of tweet, stop words, and word2vec, which was combined with deep learning approaches of Long Short-Term Memory and Convolutional Neural Network. The models that were trained trained employed in dataset

of IMDB, containing 50,000 reviews of movies. The optimizer could recognize the reviews of movies and rate reviews of tweets with accuracy values of 87.74%. The study has not mentioned that how issues related to noise have been dealt with. For this reason, low values have been accomplished, which represent the importance of using preprocessing stages.

Meena et al.²⁸ suggested an automated classification approach that employed CNN. The main purpose of the study was to make sure if the tumors are shown in the images. The Br35H benchmark dataset has been utilized for doing the experiment, which is publicly accessible. Before starting the training stage, the data were augmented to decrease consumption of time and improve accuracy. It was compared with other models, and it was represented that it could outperform the other models. A limitation of the study is its lack of diversity regarding dataset. Thus, it has to be tested with some other datasets to represent its accuracy while being measured using other datasets. Therefore, it can be asserted that the accuracy value can have fluctuation.

Meena et al.²⁹ combined two diverse networks for POI suggestion. The initial model employed BiLSTM (bidirectional long short-term memory) for sentiment prediction and was trained on a dataset related to elections. It has been noted that the suggested model surpassed existing ones in performance. Subsequently, this model is applied to the Foursquare dataset to determine class labels. Following that, embeddings for users and points of interest (POI) were created. The subsequent model recommended top POIs and their associated coordinates to users by utilizing the LSTM. Filtered user locations and interest have been utilized to suggest POIs from the Foursquare dataset. The outcomes of the suggested sentiment analysis-based POI recommendation system were compared with various state-of-the-art methods and were quite positive.

Meena et al.³⁰ developed a study in which deep learning developed on the basis of transfer learning. The model could help medical experts and other people determine if they are infected with Monkeypox. Moreover, InceptionV3 was employed in the present article that has been trained using Monkeypox dataset, which available on internet. However, this model was evaluated by utilizing only one dataset, which can decrease the reliability of achieved accuracy value.

The motivation of the present study is that CNN has been used, which is mostly employed for classification tasks. In other studies that have been mentioned in the literature, only neural network was used; therefore, they could not achieve desirable results and efficiency. In this study, Enhanced Gorilla Optimization Algorithm has been employed that fully tried to optimize the hyperparameters of CNN. However, if raw data get inserted into neural network, the results will not be that proficient. For this reason, several preprocessing stages have been carried out that could immensely increase the performance of the model. In the end, the suggested model CNN-EGTO was compared with other models, and the results represented that it could outperform the classification accuracy of other models. In the Table 1, all the information about the literature review discussed above have been provided.

Research questions

In each article, there should be questions, which should be answered. In fact, the questions are foundations of developing a study. In this section, there will be 2 questions which will be answered in the result section. The questions are as follows:

1. Does Enhanced Gorilla Troops Optimization algorithm can surpass other algorithms in efficiency?
2. Does the combination of CNN and EGTO outperform the other models?

Database

The model has been trained employing two various datasets that have been found to be datasets of twitter developed via SemEval-2016. The initial dataset, called SemEval-2016-1, is composed of 3694 number of tweets; moreover, the dataset two, called SemEval-2016-2, is composed of 1122 number of tweets. All of these tweets have been determined as positive or negative in a manual manner. In other words, there are 2 diverse datasets that have been employed in the present study. The datasets have been already labeled as positive or negative. Then, the results of the model will be determined based on them. The site can be accessed by the following link: <http://alt.qcri.org/semeval2016/>.

Author	Approach	Dataset	Accuracy	Limitations
Jianqiang et al. ²⁴	CNN	Twitter	92.3%	1 lack of capture long-range context or dependencies 2 Inability in comprehending full of a tweet or sentence
Umer et al. ²⁵	CNN-LSTM	1 Hate speech 2 Women's e-commerce reviews of clothing 3 US airline sentiments	90.26%	1 High complexity 2 Time-consuming 3 Expensive
Tam ²⁶	ConvBiLSTM	1 SST-2 datasets 2 retrieved Tweets	91.13%	Lack of information about how to deal with abbreviations and so in data
Gandhi et al. ²⁷	LSTM-CNN	IMDB	87.74%	Lack of information about how to deal with noise in data
Meena et al. ²⁸	CNN	Br35H	98.99%	1 Lack of diversity 2 Unreliability of accuracy value
Meena et al. ²⁹	BiLSTM	Foursquare	Was not mentioned	No data about accuracy value
Meena et al. ³⁰	InceptionV3	Monkeypox	91.85%	Lack of dataset diversity

Table 1. All the data relevant to studies in literature review.

Dataset	Number of tweets	Positive	Negative
SemEval-2016-1	3694	3054	643
SemEval-2016-2	1122	832	290

Table 2. Label distribution.

	Dataset	
	SemEval-2016-1	SemEval-2016-2
Number of users	3536	2198
All the tweets	774,244	491,902
Average number of tweets each day	218.96	223.80

Table 3. User distribution.

Generally, SemEval-2016 Task 1 has been developed for Semantic Textual Similarity purpose. However, in this study, the aim is to use it for sentiment analysis tasks. The dataset comprises different pairs of sentences gathered from diverse sources, such as news articles and social media content. Moreover, second dataset is SemEval-2016 Task 2 which has been particularly designed for sentiment analysis of tweets.

This assignment is centered on identifying the sentiment conveyed in tweets, particularly regarding their polarity (negative, positive, or neutral) and the sentiment directed at specific subjects. The tweets included in these datasets address a variety of topics, such as Political Figures that discussed sentiments towards personalities like Hillary Clinton and Donald Trump; Social Issues that discusses conversations around movements like feminism and topics such as climate change and the legalization of abortion; and General Sentiment Topics that discusses the datasets feature tweets that articulate perspectives on diverse societal issues, reflecting public sentiment on these matters.

The aforementioned datasets have IDs of tweets and their comment, which have been categorized as negative or positive. There are several keywords listed in the current study, including flue, fever, break up, happiness, sadness, and so on, to filter and label the tweets as negative and positive. In fact, different states of users can influence polar of the sentences. They can be downloaded by employing the Twitter API. The subsequent table displays some data relevant to two datasets of Twitter that have been utilized in the present study (Table 2).

In every dataset, several users posted the tweets has been developed. After that, public profiles and timelines of users have been saved. A considerable quantity of unlabeled tweets was provided, and the table above provided the number of sentence in each datasets and the number of positive and negative sentences in each one. The subsequent table provides some information relevant to the average number of tweets per user, all the tweets, and number of users (Table 3).

The findings of all experiments are presented employing 10-fold cross-validation. This method involves dividing the dataset into 10 subsets, using one subset for training the classification model, and the other nine subsets for testing the model. This process has been repeated 10 times, and the validation findings have been then averaged after the 10 iterations. There are several challenges of the dataset employed in the article. Tweets usually have abbreviations, slangs, and informal language that make it challenging for the network to handle and classify the data effectively. However, several preprocessing stages have been conducted in order to prevent from this issue.

Features

The first phase of the process entails extracting pertinent features from the dataset, which is vital for preparing the data for prediction or classification tasks. Prior to this extraction, it is important to properly identify and recognize the features that are most relevant to the issue. This identification frequently involves examining the data to ascertain which attributes or variables will be beneficial in developing effective predictive models. After the relevant features have been determined, the required steps and actions can be taken.

The process of feature extraction is a vital stage in preparing the dataset for sentiment classification. It guarantees that significant information is derived from raw text, allowing the model to reliably classify tweets as either positive or negative. In this research, 31 unique features were extracted. These features were chosen for their capacity to capture the linguistic, syntactic, and semantic characteristics of tweets, which are crucial for sentiment analysis. In the present section, 31 features have been employed for training the suggested network. All these features have been listed down in Table 4.

Number of tweets of a user

The current attribute indicates the quantity of tweets of a user that have been shared by Twitter API. Twitter enables to return an assortment of tweets that have been recently shared by a user. Merely 3200 tweets of each

Features ID	Feature Description
F1	A user's tweets number
[F2, F3, F4]	Negative and positive tweets number posted by a user.
[F5, F6, F7]	Mean number of positive and negative tweets posted by a user
[F8, F9]	Adjectives number and their mean
[F10, F11]	Nouns number and their mean
[F12, F13]	Adverbs number and their mean
[F14, F15]	Verbs number and their mean
[F16, F17]	Hashtags number and their mean
[F18, F19]	Mentions number and their mean
[F20, F21]	URLs number and their mean
[F22, F23]	Emoticons number and their mean
[F24, F25]	Question Marks' number and their mean
[F26, F27]	Exclamation marks number and their mean
[F28, F29]	Positive words number in SemEval-2016.
[F30, F31]	Negative words number in SemEval-2016.

Table 4. Features utilized in the present article.

user might be returned. This feature can get activity level of each user on Twitter platform. In this case, the users with high activity might represent different semantic patterns.

Highly active users may exhibit distinctive patterns of sentiment.

Number of negative and positive tweets posted by a user

The purpose of this features is to evaluate the overall feature of a user. The words produced by a user can determine his or her personality. As a result, the tweets have been determined on the basis of 3 polarities, including positive, neutral, and negative. To accomplish this purpose, we have to gather tweets that have been shared by the users involved in the present study. Generally, this can help measure entire sentiment disposition of each user.

Hence, an application has been created that can receive output of SentiStrength that has been described in the following. A tweet has been classified as positive if positive strength of sentiment exceeds its negative sentiment strength and 1. Else, it has been classified as a negative tweet. If a tweet receives the value of 1 for positive and negative sentiment strength, it has been taken into account as neutral. A tweet may be classified as neutral once it has equal scores in strength of negative and sentiment positive, such as a tweet that has the values of 3 for strength of negative and 3 for positive. The current procedure has been computed in the following way:

$$SS(t_i) \begin{cases} \text{Positive} & \text{if } SS(t_i)_{pos} > 1 \text{ and } SS(t_i)_{pos} > SS(t_i)_{neg} \\ \text{Negative} & \text{if } SS(t_i)_{neg} > 1 \text{ and } SS(t_i)_{neg} > SS(t_i)_{pos} \end{cases} \quad (1)$$

Considering the users, u is involved in $U = [u_1, u_2, \dots, u_n]$. Here, the user net has been illustrated by U , and the quantity of users that have been extracted has been displayed via n . Moreover, the function that takes the input has been depicted via $SS(t_i)$, where t_i is involved in T_u , and it determines the label of the tweets on the basis of the SentiStrength algorithm. The values of the negative and positive sentiment strengths have been demonstrated via $SS(t_i)_{neg}$ and $SS(t_i)_{pos}$. To do so, the following equation is employed.

$$SS(t_i) = \begin{cases} \text{Positive} & \text{if } SS(t_i)_{pos} > 1 \text{ and } SS(t_i)_{pos} > SS(t_i)_{neg} \\ \text{Negative} & \text{if } SS(t_i)_{neg} > 1 \text{ and } SS(t_i)_{neg} > SS(t_i)_{pos} \\ \text{Both} & \text{if } SS(t_i)_{pos} = SS(t_i)_{neg} \neq 1 \\ \text{Neutral} & \text{otherwise} \end{cases} \quad (2)$$

Number of adjectives, nouns, adverbs, and verbs

Each feature in this category has been computed on the basis of each tweets shared by a user. The features have been found to be great indicator to determine the polarity that has been employed. For instance, if there are myriad adjectives within a tweet, the polarity can be determined easily³¹. For example, when a user posted "this night is hateful and full of darkness", the adjectives "hateful" and "darkness" are negative adjectives.

Number of hashtags

Each hashtag has been employ to depict the pertinence of tweets to a predetermined issue. They have been made by users, which is employed to gather the tweets shared. If the purpose is to make hashtag, the symbol # should be employed before the topic.

It has been mentioned that employing hashtags within the set of feature has two purposes. One of the reasons is that they can serve as a great indication of a user expressing their opinions and thoughts on the hashtag's topic³². Another object is each hashtag can be utilized to identify spam tweets; in fact, a study demonstrated

that each hashtag has been classified as spam once its frequency of tweet has been found to be high³³. Hashtags usually illustrate opinion of users, like #GreatSunnyDay, which can aid in determining polarity of text.

Number of mentions

These have been utilized to specify the individuals who receive a tweet. Additionally, each user is also able to directly refer to other individuals to capture their attention via including an explanation such as @Username.

Number of URLs

The entire quantity of URLs of the users can be counted by this feature, which has been computed like the features of emoticons by employing Eq. (3).

Number of emoticons

Each emoticon has been considered a visual illustration of a facial expression created employing the features found on a conventional keyboard. When included in a tweet, it serves as a reliable sign of the emotional condition of writer. It was emphasized that the existence of an emoticon most of the time expresses the original sentiment. Moreover, emoticons have been found to be strong indicators of polarities and sentiment of a sentence, which can help the network perform better. As a result, this characteristic is included in the vector of feature and is computed in the following manner:

$$E(T_u) = \sum_{i=1}^{|T_u|} |[t_i \in T_u : HasE(t_i) \neq 0]| \quad (3)$$

where, the aggregation function has been demonstrated by $E(T_u)$ that captures as the input, displayed via T_u ; moreover, it has all the tweets posted by user u within the set of user U , then it returns the entire quantity of emoticons. The quantity of tweets of a user has been depicted via $|T_u|$, the function that gets employed in tweet t_i has been indicated via $HasE(t_i)$, which can return the quantity of emoticons within a tweet. Tokenizer of twitter NLP has been employed for identifying the emoticons. Furthermore, the mean quantity of a user's question mark has been calculated. The tokenizer can be accessed by clicking the following link: <http://www.cs.cmu.edu/~%20ark/TweetNLP/>.

Number of question marks

The entire quantity of question marks has been computed, employing the equation illustrated below:

$$QM(T_u) = \sum_{i=1}^{|T_u|} |[t_i \in T_u : HasQM(t_i) \neq 0]| \quad (4)$$

The aggregation function has been demonstrated via $QM(T_u)$, containing the tweets of u^{th} user within the set U ; furthermore, the entire quantity of the question marks gets returned. The entire quantity of tweets of a user has been represented by $|T_u|$, and $HasQM(t_i)$ has been found to be a function that has been employed in tweet t_i and can return the quantity of question marks within each tweet. Eventually, the mean quantity of question marks of each user can be calculated.

Number of exclamation marks

This shows the entire number of marks of exclamation that occurs within tweets of the users, which has been computed like feature of question marks in Eq. (4).

Number of negative and positive words in SemEval-2016

Here, the features of lexicon have been employed by being extracted from lexicons, which are available in public and prove that it is a strong feature³⁴. People write in a different language style in comparison with other texts. As a result, the SemEval-2016 dataset has been employed. By utilizing the lexicons from the dataset, terms such as "happy" and "sad" were recognized to capture clear sentiment signals.

Methodology

Implementation procedure

The objective of the present study is developing a robust model for analyzing tweets sentimentally, particularly recognizing negative and positive sentiments that have been declared. For preparation stage, the datasets were collected and preprocessed from two datasets that ensured labels of tweets as negative or positive. In the preprocessing stage, the features were extracted and all the noise and unwanted characters, like punctuation, usernames, and URLs, from the data were removed. Then, relevant features, like hashtags, number of adjectives, and mentions, were extracted. After that, the data got ready to be inserted into the neural network to accomplish high efficiency and performance. Moreover, the authors the suggested model by integrating Convolutional Neural Network (CNN) with Enhanced Gorilla Troops Optimization Algorithm (EGTO) with the purpose of optimizing network's hyperparameters.

Word Embedding

This technique receives raw input in the initial stage, then it decomposes the input into tokens or words. The tokens get altered into numerical vectors. GloVe technique has been used to generate word vectors' matrix. This technique has been employed to evaluate the efficiency of model. The text with n number of words is represented via T that is equal to $[w_1, w_2, \dots, w_n]$. In the following, each word gets transformed into word vectors that each of them is d -dimensional. The input is mathematically illustrated subsequently:

$$T = [w_1, w_2, \dots, w_n] \in R^{m \times d} \quad (5)$$

The input text must maintain a consistent length, represented by (l) , due to their varying sizes. Additionally, the zero-padding method has been used to extend its length. Any text that exceeds the predetermined length l will be reduced. Conversely, when the text is shorter than l , the zero-padding technique is applied to achieve the required length. Consequently, all texts will form a matrix with uniform dimensions. Each text with the l dimension can be represented in the subsequent manner:

$$T = [w_1, w_2, \dots, w_n] \in R^{l \times d} \quad (6)$$

Convolutional neural network (CNN)

Generally, Neural network can be used for different purposes, including Predicting Coronary Heart Disease³⁵, Improving weak queries³⁶, Unified Topic-Based Semantic Models³⁷, and Cardiovascular disease diagnosis³⁸. Also, Convolutional Neural Network (CNN) has been utilized in the present study in order to do the task of sentiment analysis. Moreover, there are some other models, which have been used for doing sentiment analysis task³⁹.

The feature of the tweets must be extracted within a manual manner to classify the tweets by getting help from learning techniques⁴⁰. Several approaches of tweet classification have been used, and the experts of the major have been consulted to complete the present assignment. The extracted attributes relevant to the tweets are used as an intelligent classification method using neural network or as input for training a model.

A CNN (Convolutional Neural Network) has been considered a sort of ANN (Artificial Neural Network) involving weights, neurons, and weights like other kinds of neural networks. On the other hand, the major distinction between CNNs and ANNs is their particular skill in deep learning. It allows them for classifying diverse tweets.

CNNs have been found to be a sort of totally multilayer and linked neural network, including fully linked layers, pooling layers, and convolutional layers. A fundamental neural network receives and passes the input to some hidden layers. The hidden layers own some neurons in the structure of vector, each of which has meticulously linked to the neurons of the prior layer. The neurons of the layers can work in an independent manner and have not had common weights and connections.

The layer of output within the current model has responsibility of making the ultimate decisions relevant to the data category. Regarding CNN, the input has been found to be a multi-layered or a depth matrix that has been named a tensor. The input has been in the structure of a tweet, and the model's layers have not been similar to the normal ANN, since they include neurons that have 3 dimensions.

Regarding the suggested model, each neuro of a layer moves to a small region within earlier layer that have not been linked to the neurons of the earlier layer. Within a normal NN, the neurons are linked. But the neurons are three-dimensional within a CNN (Fig. 1).

Convolutional layer

The primary layer in this sort of neural network is composed of convolutional neural networks, which extract features through the application of a convolution operation on the input data, generating feature maps as output. Each neuron within a feature map is equipped with a set of features that enable classification of tweets, as positive, negative, and neutral^{41,42}. These associations are made within small, local regions, with neurons within the starting hidden layer being connected to a specific space of the input neurons.

The current minor area is called local perception space of convolution^{43,44}. Additionally, the convolutional output of the layer neuron in diverse depth (m), length (h), and width (w) are expressed in the following manner:

$$y_{w,h,m} = f \left(\sum_{j=(w-1)S+1}^{(w-1)S+K} \sum_{i=(h-1)S+1}^{(h-1)S+K} b_m + \sum_{k=1}^N W_{k,m} x_{j,i,k} \right) \quad (7)$$

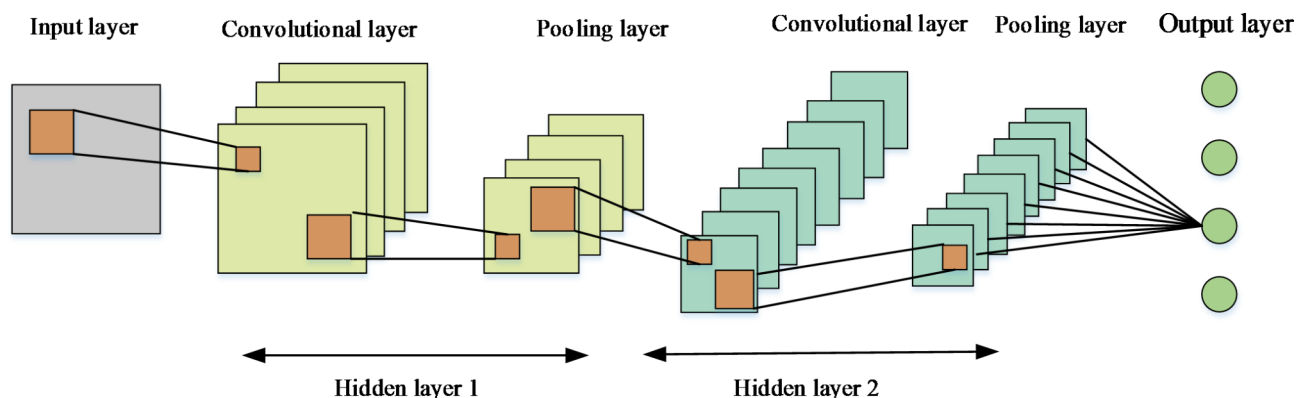


Fig. 1. The structure of Convolutional Neural Network.

where, the input of location j , k , and i are illustrated via $x_{j,i,k}$, the neurons' 5×5 normal weights are displayed via $W_{k,m}$, the activation function has been represented via f , and the typical bias of the neurons are displayed via b_m .

The utility of activation functions within the layer of convolution enhances the non-linear characteristics of the output. As a result, every neuron in the first hidden layer demonstrates similar attributes in different areas of tweet classification. The feature map represents the output of the input layer's neurons within the hidden layer.

The dimension of output matrix of $W_2 \times H_2 \times D_2$ of the convolutional layer are accomplished by input matrix of $W_1 \times H_1 \times D_1$. The parameters H_2 and W_2 have been explained below:

$$W_2 = \frac{W_1 - F + 2P}{S + 1} \quad (8)$$

$$H_2 = \frac{H_1 - F + 2P}{S + 1} \quad (9)$$

where, D_2 illustrates the quantity of filters, zero padding has been displayed via P , kernel has been displayed via F , and step size has been represented via S . The equations display that the filters have $D_1 \times F \times F$ weight and biases in accordance with the K quantity of filters, and there will be generated K quantity of $(D_1 \times F \times F)$ weights and biases. It displays more quantity of parameters with the purpose of training convolutional layers.

Pooling layer

A fusion layer has been commonly placed after convolutional layers. The current layer's goal purpose is reducing the quantity of variables that must be trained. This decrease in parameters assists in declining the amount of computation necessary throughout training and can control probable meta-learning of the model. The current layer has been utilized in all layers of the input's depth, and it modifies the input size. This layer owns 2 typically utilized activation functions, including average and maximum pooling. The primary method, maximum pooling, has been more commonly utilized.

The maximum combination approach chooses the adjectives in all windows and can convert them to the output⁴⁵. The windows move the tweets identical to convolution's function from the side to the bottom side and from the left side to the right side, enjoying a particular step size. After that, it can send the finding to the output^{46,47}. Since the current procedure is utilized in all the tweets, the output's depth is identical to the input layer's depth that remains the same. The link between output tensor with the dimension of $W_3 \times H_3 \times D_3$ accomplished via input tensor's step of convolution with the scope of $W_2 \times H_2 \times D_2$ has been explained in the following manner:

$$W_3 = \frac{W_2 - F_P}{S_P + 1} \quad (10)$$

$$H_3 = \frac{H_2 - F_P}{S_P + 1} \quad (11)$$

$$D_3 = D_2 \quad (12)$$

here, the kernel size is illustrated via F_p , and the step size of the pooling layer is indicated via S_p .

FC layer

The eventual layer within the CNN has been found to be Fully Connected Layer. Identical to the typical Neural Network, it can produce a thorough link with the output of the prior layer. The current layer takes an input and can produce a vector with N elements considered output. The number of the categories that the intended model can categorize equals N . The constructed model produces a vector that is composed of multiple elements once a tweet has been inserted as an input.

It is essential to take into account the links with the purpose of assessing learning or categorization accuracy while utilizing learning optimizers. The link has to decrease to the least amount and must be optimized by other suitable techniques⁴⁸⁻⁵⁰. Therefore, the fundamental concept of the learning network training is to decrease the loss function to the least amount while moving through suitable phases to obtain better outcomes. Since the quantity of parameters trained has been found to be highly fantastic and the current models have deep layers within the CNNs, the models lead to overlearning during training stage.

Numerous solutions have been suggested to solve the current problem. A pioneer method can increase the amount of the data of training. Moreover, the other approach tries to add a proper dropout layer within the fully connected layer in order to overcome the current issue.

Objective function

Selecting the hyperparameters is one of the greatest issues in optimization procedure. The main goal of the fitness function is to decrease the error existing between the network's output and the necessary output to the least amount by identifying the finest to utilize them as decision variables. Hence, the fitness function is the network's accuracy.

In the end, the major aim is to utilize a method of optimization to increase the accuracy value of the suggested model, which has been computed in the subsequent manner:

$$\max_{\vec{H}_p \in R^i} CNN(\vec{H}_p, \vec{W}, Td_i) \quad (13)$$

where, the hyperparameter vector of i^{th} dimension is illustrated via H_p , trained data is demonstrated via Td_i , i is smaller than i_{max} , the user-defined parameter is displayed by i_{max} that can strike a balance between complexity and accuracy, and the vector of weight is depicted by \vec{W} .

In the end, a superior value of the current model signifies a high extent of complexity, whereas a low value demonstrates low degree of accuracy, depicting the fact that a suitable framework of this model highly relies on the suitable choice of i_{max} . The suggested technique does sentiment analysis by employing a particularly made convolutional network to analyze the polarity of the tweets and data.

Gorilla troops optimization (GTO) algorithm

The Gorilla Troops Optimization is a novel metaheuristic algorithm, which has been inspired by the gorillas' behavior in the group. This algorithm includes two statistical steps, including exploitation and exploration. Moreover, the process of optimization employs five different operators that involve exploration and exploitation, which are on the basis of the manners of gorillas. Through the global search stage, 3 diverse procedures have been utilized, and their application is defined subsequently:

Movement to indefinite location is a crucial technique in order to improve the exploration of GTO. A superior ratio of exploration to exploitation can be attained by approaching other candidates. The ability of GTO for investigating other optimization spaces has been also considerably improved by approaching to other gorillas. The steps involve approaching to a specific location, exploration of the area, and search for opportunities of optimization. Utilizing the two operators through the exploitation step results in an essential enhancement in the search effectiveness. Ultimately, utilizing GTO for searching data is defined subsequently:

- The GTO optimization algorithm has three possible responses; if any of them is better than the present one, it will be used. The gorillas' location vector is depicted by O and the location vector of the gorilla agent is shown at every stage by XO . The best solution found after each iteration is kept as the silverback.
- During the optimization procedure, only one silverback stands out for the number of search parameters selected from the whole group.
- The gorilla community has been simulated using three different components: XO , O , and Silverback solutions.
- Gorillas can improve their robustness by combining to a superior group or identifying superior food resources. Subsequent to each iteration, the GTO algorithm indicates the generated solution by XO . When the recently identified solutions are not distinctive, the present solution stays in the memory XO , and if not, the newly identified solution replaces the present one in XO . Gorillas mostly live in groups, so they are seldom alone. So, they reside in communities and are guided by a dominant silverback who makes decisions for the whole group. The candidates also take part in hunting activities. In the algorithm, the weakest gorilla is taken into account as the least solution, and they are usually avoided by the others. On the other hand, getting closer to the silverback, who is the most superior, can enhance the situation for all the gorillas.
- The GTO algorithm is broadly useful in solving optimization issues as it integrates the fundamental principles of group living among other candidates and their search for food. It employs a variety of optimization strategies to accomplish its goals.

Exploration

Investigating the processes used in the initial stage of GTO exploration is the primary stage. Gorillas are social candidates that reside in groups and are led by a silverback. However, under specific situations, a gorilla may opt to depart from its group. Once they have left their group, gorillas go to various places, some of which they may recognize, while others may be new and unfamiliar territory. The optimization process for gorilla groups takes into account each gorilla as a solution, and the top gorilla of each step is called the silverback. Three different techniques are used for the exploration phase, including transferring to other individuals, relocating to an acquainted space, and going to a new location. Each of these approaches is selected on the basis of a thorough strategy.

The process of emigration to an unknown location is illustrated by b . When $rand$ is lower than b , the first procedure has been developed, and when $rand$ is higher than or equivalent to 1, the movement towards the candidates' direction has been selected. Based on the employed procedure, all the procedures lead to the extensive capability of the GTOA. The initial stage may consider the complete search space. The effectiveness of exploration in the second process is improved, and as a result, the 3rd strategy prevents GTO from becoming trapped in locally optimum situations. For the global search stage, the three utilized processes have been mathematically defined in the following way:

$$XO(t+1) = \begin{cases} (\bar{O} - O) \times m + O, & rand < d, \\ (r_2 - S) + Z_r(t) + C \times F, & rand \geq 0.5, \\ O(t) - C \times (C \times O(t) - XO_r(t)) + r_3 \times (O(t) - XO_r(t)), & rand < 0.5, \end{cases} \quad (14)$$

here, the location of the individual in the t^{th} iteration has been indicated by $XO(t+1)$. The present location of the individuals has been depicted via $O(t)$. The stochastic variables r_1 , r_2 , r_3 , and $rand$ are in the range of 0 and 1. The possibility of selection of an unidentified place is indicated by b , it ranges from 0 to 1, and

it is provided before the procedure of the optimization. The variable O indicates the parameters' minimum boundary and the variable \bar{O} demonstrates the parameters' maximum boundary. Two organs of candidates stochastically selected from the whole group are demonstrated by O_r and XO_r . One location vector of the gorillas is selected in a random way and involves the updated locations at every step. The variables F , C , and S can be calculated using the following formulas:

$$S = F \times \left(1 - \frac{It}{MaxIt}\right) \quad (15)$$

$$F = \cos(2 \times r_4) + 1 \quad (16)$$

$$C = S \times c \quad (17)$$

here, the variable It demonstrates the amount of the current iteration. The maximum quantity of the iterations for carrying out the process of optimization is indicated by $MaxIt$. The variable cos illustrates the cosine function. The variable r_4 is a random number that ranges between 0 and 1, and it is upgraded after every iteration.

The optimization procedure generates values with notable changes over an extended period at the beginning, but these changes decrease as the procedure nears final. A random value represented by c and in the range of -1 to 1 is used to compute parameter C . This equation is applied to define silverback leadership. In the early stages of leading a group, the silverback gorilla may make mistakes in managing the group or finding nutrition because of its lack of experience. However, it learns experience over time and establishes steady leadership. The variable F has been computed utilizing the formula below. The parameter y is a stochastic quantity in the dimension of problem and it ranges between $-S$ and S :

$$F = y \times Z(t) \quad (18)$$

$$y = [-S, S] \quad (19)$$

After completing the exploration phase, the group optimization process is finished. The cost of each XO has been calculated at this step and the solution $XO(t)$ has been utilized instead of $O(t)$ if $XO(t) < O(t)$.

Exploitation

Through the exploitation phase of the GTO algorithm, two behaviors have been carried out, including looking for the silverback gorilla and competition for the mature couples. The silverback carries out various duties, including managing the group, making all decisions, coordinating the group's movements, and guiding the troop to food sources. It is responsible for the safety and welfare of the group, and all the candidates in the group follow its leadership. In the event of a disagreement between the silverback and other male gorillas, the candidates can assume the management of the group. When the silverback becomes older and passes away, the blackback gorilla can take charge of the troop. The two procedures employed during the exploitation step can help determine whether to follow the silverback or to compete for mature females. By adhering to the two procedures utilized through the exploitation phase, it is possible to ascertain whether it is superior to compete for mature females or to follow the silverback. If $S \geq Q$, the decision has been made to follow the process of silverback. If $S < Q$, the decision has been made to compete for the adult mature.

Following the silverback

The silverback candidate is in excellent condition and is relatively young. It is trying to form a new troop of gorillas. The other male gorillas in the troop are also youthful and thus adhere to the directives of the silverback. They obey his orders, such as going to various areas for searching food resources. Each potential group member contributes effectively to the general movement of the troop. If $S \geq Q$, this procedure has been utilized, and it has been statistically defined as follows:

$$XO(t+1) = C \times M \times (O(t) - O_{silverback}) + O(t) \quad (20)$$

$$M = \left(\left| \frac{1}{N} \sum_{i=1}^N XO_i(t) \right|^g \right)^{1/g} \quad (21)$$

$$g = 2^C \quad (22)$$

here, the variable $XO_i(t)$ demonstrates each gorilla's location vector in the t iteration. The total number of candidates is indicated by N . The variable $O_{silverback}$ represents the silverback's location vector (the best solution). The variable $O(t)$ illustrates the gorilla's location vector.

Competition for mature females

If $S < Q$, the exploitation phase's second process has been chosen. As young candidates grow up, they try to increase their group by choosing a female. This procedure frequently includes clashes between the males, which may endure for multiple days and include other individuals of the group. In order to measure the current manner, the next formula has been utilized:

$$XO(i) = O_{silverback} - (O_{silverback} \times W - O(t) \times W) \times A \quad (23)$$

$$W = 2 \times r_5 - 1 \tag{24}$$

$$A = \rho \times E \tag{25}$$

$$E = \begin{cases} N_1, rand \geq 0.5 \\ N_2, rand < 0.5 \end{cases} \tag{26}$$

here, the variable $O_{silverback}$ illustrates the silverback's location vector. The gorilla's location vector is demonstrated by $O(t)$. The variable W demonstrates a model for the impact force. The variable r_5 is a stochastic amount that ranges between 0 and 1. The variable A is a coefficient vector to ascertain serious violence's degree. The variable E has the responsibility of constructing a model for simulation of the viciousness impacts on the solution dimensions. The variable ρ is computed before optimization. If $rand \geq \frac{1}{2}$, the variable E will equal the random numbers. Nevertheless, if $rand \leq \frac{1}{2}$, the variable E will be ascertained by a stochastic number in the normal distribution. The E variable's amount is reliant on the dimensions of the issues and normal distribution. The group organization's process is ended at the end of the exploitation phase. So, fitness of each XO solution has been calculated. If $XO(t) < O(t)$, the $XO(t)$ solution is utilized. Therefore, the best solution has considered as silverback and it has been determined in this phase. The flowchart of the suggested algorithm can be seen in Fig. 2.

Enhanced Gorilla troops optimization (EGTO) algorithm

The proposed algorithm has been enhanced in order to enhance its efficacy and strike a superior balance between exploitation and exploration within the search process. The algorithm with a constriction element and an elimination step can be enhanced to improve its strength, solution quality, and convergence rate, which can be used by metaheuristic algorithms, like the Gorilla Troops Optimization Algorithm.

- (A) Constriction element: The algorithm's use of a constriction element is essential for controlling the velocity of the search process. It allows the particles or solutions in the search space to converge towards potential areas more rapidly while also exploration of other areas. The constriction element enhances the acceleration coefficients according to the swarm's optimal performance, and it results in preventing excessive movements and promoting faster convergence. The constriction element has been applied to the random variables, including r_1, r_2, r_3 , and $rand$ utilizing the following equation:

$$r_1 = \frac{\theta}{1 - \sqrt{\varphi^2 - 4\varphi}} \tag{27}$$

$$r_2 = \frac{\theta}{1 - \sqrt{\varphi^2 - 4\varphi}} \tag{28}$$

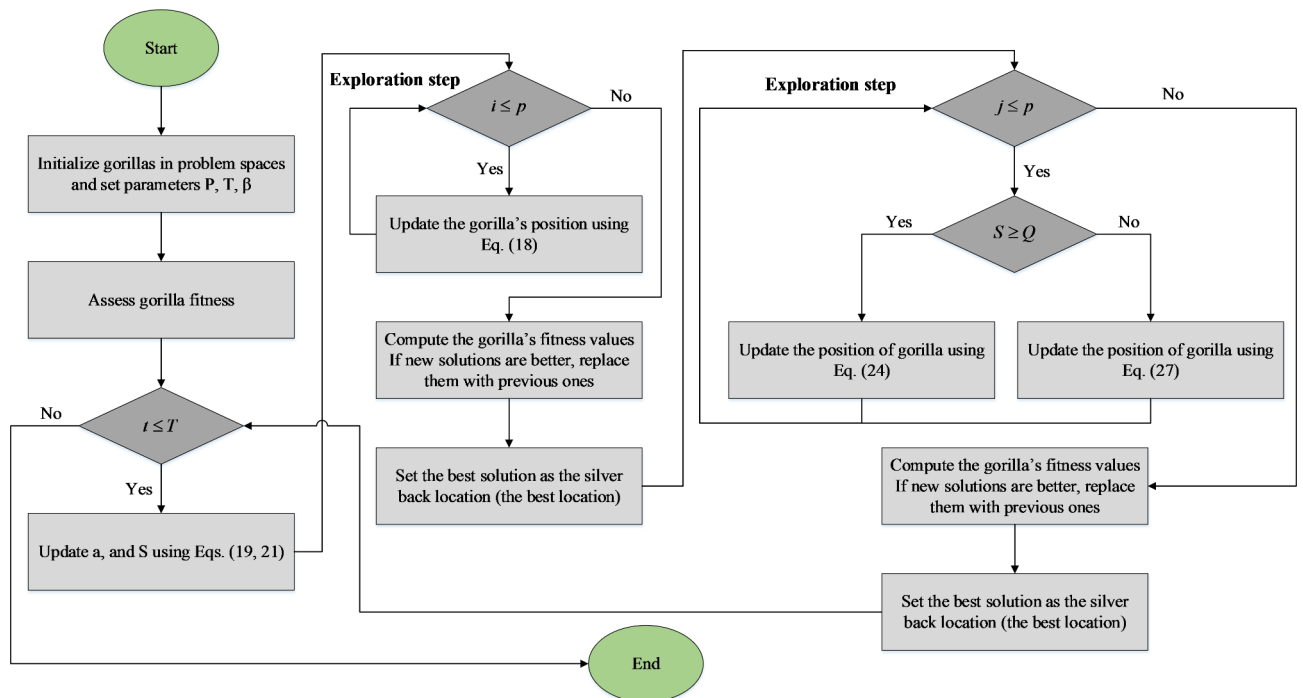


Fig. 2. The flowchart of the suggested algorithm.

$$r_3 = \frac{\theta}{1 - \sqrt{\varphi^2 - 4\varphi}} \quad (29)$$

$$rand = \frac{\theta}{1 - \sqrt{\varphi^2 - 4\varphi}} \quad (30)$$

here, the variable $\varphi \geq 4$, and the variable $\theta \leq 1$.

Based on the study's findings utilizing constraint element across all dimensions is a promising approach in all dimensions. In a prior research⁵¹, it has been shown that this specific technique offers greater findings compared to other akin methods. The current study presents a novel enhancement of the constraint factor-based method that has been found to be a validated technique in scientific investigation. Instead of keeping their values constant, the proposed enhancement suggests a gradual, linear reduction in the variables r_1 , r_2 , r_3 , and $rand$. To carry out this approach, the θ value can be enhanced in an iterative way utilizing the formula below:

$$\theta_j = \theta + \left(\bar{\theta} - \theta \right) \times \frac{L - j}{L - 1} \quad (31)$$

here, the variable L indicates the highest amount of the iterations. The variable j demonstrates the current iteration within the procedure. In order to maintain the computational stability of the EGTO algorithm, it is crucial to identify a constriction factor.

Using a constriction factor to strike a balance between exploitation and exploration is an important part of improving metaheuristic algorithms, like the Gorilla Troop Optimization Algorithm. This enhancement helps the algorithm to more rapidly converge towards the optimal solutions. The constriction factor enhances the acceleration coefficients based on the swarm's best effectiveness, and enables the solutions or particles to converge more efficiently towards promising regions as they explore other areas. Apart from enhancing convergence, this improvement also boosts the algorithm's robustness. Modifying the acceleration coefficients appropriately can prevent premature convergence and prevent being trapped in local optimum that is a common challenge in complex and dynamic problem landscapes. By integrating these improvements, the optimization algorithm, which is inspired by gorilla groups, can efficiently address complicated optimization problems and provide more accurate and reliable solutions.

(B) Elimination phase: After every iteration, a process known as the elimination phase is carried out to remove the least effective solutions or candidates from the group. This phase eliminates a segment of the group based on particular selection criteria, like fitness value. By removing less strong solutions, their influence on the search is reduced, and enables superior solutions to subsequent iterations.

It is essential to establish a set of criteria for identifying the solutions that illustrate the lowest performance in order to statistically include the elimination phase in the algorithm. Furthermore, it is necessary to create an effective procedure for removing these inferior solutions from the population. The incorporation of the elimination phase into the algorithm can be defined as follows:

- Initialize population (P) with individuals (N)
- Evaluate each individual's fitness employing the cost function.
- Set the maximum amount of iterations or termination situation (T)
- Set the elimination rate (ER) that demonstrates the proportion of the solutions that should be eliminated in each iteration
- Set the elimination criteria, such as selection of the solutions according to the fitness value or other criteria.

After every iteration of this algorithm, the elimination phase procedure has been implemented. This process includes categorizing the population according to their fitness values and then removing a specific number of solutions with the lowest effectiveness (NE), as defined by the elimination rate (ER). After this, additional actions, such as replacement or reproduction, are carried out to maintain the desired population size, thereby enabling the population to advance toward improved solutions over time.

The algorithm utilizes the elimination phase as an iterative procedure in its statistical equation, allowing for the exclusion of less effective solutions and the advancement of better candidates in subsequent iterations. The particular application specifics, such as the criteria for selection and methods for replacement, can differ based on the specific problem and the algorithm's design decisions.

Integrating the improvements enhances the algorithm's optimization performance, convergence speed, and adaptability to complex problem landscapes. Striking a balance between exploration and exploitation, the constriction factor is helpful, while the elimination phase assists in removing weaker solutions to enhance the evolution of superior individuals. The elimination phase also enhances solution quality by eliminating inadequate solutions, which gradually improves the general population quality. Additionally, the maintenance of diversity is ensured by eliminating poor solutions that helps in avoiding premature convergence and preserving variety within the population. This enables a more thorough exploration of the solution space and prevents the exploration of the search space from stagnation.

Results

General overview

Initially, it should be mentioned there several important stages which have been mentioned in the following. However, this subsection tries to give an overview of how the parameters of the models have been compared with each other. The parameters of the suggested EGTO have been compared with other algorithms mentioned before through evaluating their efficiency in diverse optimization problems. The EGTO has been evaluated by the use 12 benchmark functions from “CEC-BC-2017 test suite”. The current function contained hybrid, rotated, shifted, multimodal, and unimodal problems to evaluate adaptability and robustness of EGTO. There are three diverse metrics used for comparing the models, including Best values, Mean values, and Standard deviations (StD). These metrics could provide good comprehensions in terms of variability, consistency, and quality of the solutions of each algorithm. In the end, it was represented that the suggested algorithm could demonstrate decreased variability in solutions and superior accuracy. In addition, for evaluating the suggested CNN-EGTO and comparing it with other models, for evaluation metrics were utilized, including precision, accuracy, F1-score, and Recall.

Experimental setup

A. Pre-processing phase.

There exist myriad opinions in tweets relevant to the information, which have been explained in various manners via the candidates. The dataset of twitters, which have been employed within the present study, has been classified manually before. The classified dataset might have positive and negative polarity; hence, the investigation of the data gets highly easy. The raw data might be highly redundant and unstable. The data's quality influences the outcomes; as a result, the raw data has been pre-processed to enhance the quality. It tries to eliminate punctuations and repetitive words and enhances the efficacy of the data. In addition, the usernames and URLs have got eliminated and substituted with USER and HTTP, respectively. To exemplify, “I have a greaaaaaat mind” changes to “great mind”.

B. Feature extraction phase.

The enhanced dataset owns myriad diverse properties once pre-processing stage has been accomplished. The feature extraction can fully extract each aspect of the dataset. Then, the adjective has been employed with the purpose of demonstrating the negative and positive polarity of a sentence, which has been found to be really beneficial for determination of the candidates' opinions. The adjectives have been extracted, then they are segregated. The following and previous word that occurs accompanied by the adjective has been discarded. In the case of “great mind”, great has merely been extracted.

C. Classification and training stage.

Supervised learning (Machine Learning) has been found to be an essential approach to solve the categorization issues. Supervised approaches have been employed to achieve the intended outcome of sentiment analysis.

D. Evaluation stage.

In the present phase, the findings of each model has been assessed. The findings have been assessed on the basis of the most extensively employed efficacy measure within the categorization task, including F1-score, accuracy, recall, and precision. These elements have been calculated in the subsequent manner:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (32)$$

$$Precision = \frac{TP}{TP + FP} \quad (33)$$

$$Recall = \frac{TP}{TP + FN} \quad (34)$$

$$F1 - score = 2 \frac{Precision \times Recall}{Precision + Recall} \quad (35)$$

where, True Positive (TP) illustrates the quantity of the positive tweets accurately classified as positive. False Positive (FP) demonstrates the quantity of the negative tweets inaccurately classified as positive. Moreover, False Negative (FN) and True Negative (TN) have the same application like TP and FP for the negative categories. In this study, for different performance metrics have been utilized, including accuracy, precision, F1-score, and recall. Precision evaluates the proportion of true positive classification or prediction among all the positive predictions that have been done by the network that high rate of precision represents low rate of false positive, incorrect positive classifications or predictions are avoided. Recall tries to measure the proportion of true positive forecasts out of all real positive samples that high rate of recall represent that the network is able to recognize most of the real positive samples, the rate of missed recognitions is minimized. The F1-score represents the harmonic mean of recall and precision, offering a balance between these two metrics. It is especially beneficial when working with imbalanced datasets. A high F1-score signifies a favorable balance between precision and

recall. In the end, accuracy tries to measure the proportion of correct predictions, including true negatives and positives, out of the all the forecast made. High value of accuracy represent that the model can forecast massive amount of samples. However, this is not reliable where there is imbalance class.

Algorithm validation

A comprehensive validation process has been established to assess the effectiveness of this algorithm. The purpose of the present procedure is evaluating the effectiveness and strength of the proposed algorithm across a range of optimization issues with different features and levels of complexity. To achieve this, 12 diverse fitness functions are selected from the “CEC-BC-2017 test suite,” which consists of a challenging and comprehensive set of constant optimization issues. These functions cover multimodal, unimodal, shifted, rotated, and hybrid issues of diverse optima and dimensions. The proposed algorithm’s results have been tested using five different metaheuristic algorithms, each inspired by the natural behaviors of different organisms, including Gaining-Sharing Knowledge-based algorithm (GSK)⁵², Artificial Ecosystem-based Optimization (AEO)⁵³, Pelican Optimization Algorithm (POA)⁵⁴, Snake Optimizer (SO)⁵⁵, and Harris Hawks Optimization (HHO)⁵⁶ (Table 5).

Many iterations of the optimizers have been conducted on all test functions to ensure reliable and consistent results. Usually, the number of iterations has been fixed at 25, which turned out to be an extensively confirmed practice in the context of optimization. This investigation has focused on the standard deviation (StD), Best, and Mean of the performance index values produced by the optimizers. These metrics provide insights into the quality and consistency of each solution gained. In Table below, a complete assessment of the outcomes produced by the suggested EGTO have been provided in comparison to other algorithms that have been investigated.

The information depicted in Table 6 offers an intricate comparison of the outcomes achieved by the EGTO algorithm and five other state-of-the-art metaheuristic algorithms across twelve different benchmark functions. A comprehensive statistical assessment reveals the superior effectiveness of the EGTO algorithm in various optimization scenarios. In functions such as F1, F3, F4, F8, and F10, the EGTO algorithm consistently has outperformed its counterparts, displaying lower best values, means, and standard deviations. This highlights the algorithm’s resilience and efficiency in dealing with unimodal, multimodal, and hybrid problems of varying complications.

Additionally, the EGTO algorithm has displayed remarkable performance in optimization of functions, such as F2, F5, F6, and F9, which attains competitive optimal values and averages while maintaining lower variability in comparison with other algorithms. This demonstrates the algorithm’s ability to efficiently address a wide variety of optimization issues. The comparison also emphasizes the EGTO algorithm’s ability in reducing function values across various benchmark functions, demonstrating its adaptability and robustness in solving complex optimization issues.

The enhancements of EGTO, like elimination stage and constriction factor, could help the suggested algorithm avoid local optimal and strike a balance between exploitation and exploration. To exemplify, the algorithm could achieve lower values for standard deviation in F2 and F4, representing that it can produce more constant results during several iterations. Although algorithms such as HHO and SO showed impressive results on certain functions, they had difficulties as well, for example in F6 and F9. EGTO proved its adaptability by attaining competitive results in these difficult cases, highlighting its effectiveness for various types of problems.

The EGTO algorithm shows promise as a solution for addressing various optimization challenges due to its consistent delivery of competitive outcomes and lower variability. To summarize, the statistical evaluation

Gaining-Sharing Knowledge-based algorithm (GSK) ⁵²	F	0.5
	L	1
	g	20
Artificial Ecosystem-based Optimization (AEO) ⁵³	\vec{A}	1
	Temperature value (T')	400
	M	2
	f	2
	S	0.5
	l	2
Pelican Optimization Algorithm (POA) ⁵⁴	Number of prides	5
	Percent of nomad lions	0.3
	Roaming percent	0.4
	Mutate probability	0.1
	Sex rate	0.85
	Mating probability	0.4
	Immigrate rate	0.5
Snake Optimizer (SO) ⁵⁵	No. snake	50
Harris Hawks Optimization (HHO) ⁵⁶	\vec{a}	2
	\vec{r}	1

Table 5. The parameter setting of the employed algorithms.

Function	index	EGTO	GSK	AEO	POA	SO	HHO
F1	Best	0.526112	1.208525	51.74725	2.574641	1.948251	51.85198
	Mean	4.104612	9.282783	136.5538	9.649578	12.18672	104.8235
	StD	2.074361	5.58117	121.0148	7.53295	11.08378	68.69825
F2	Best	1.011473	4.69812	6.342297	6.001457	4.105919	4.233475
	Mean	22.06310	29.18423	54.51345	69.62713	55.57411	46.3904
	StD	11.26901	17.67857	44.17839	35.69611	26.19214	27.01375
F3	Best	0.01531	1.801937	19.50128	2.68536	1.013618	27.0522
	Mean	1.02192	3.51526	14.44968	4.549746	9.890121	29.37512
	StD	0.004315	2.708987	6.121794	0.018612	6.157179	14.10814
F4	Best	1.400236	3.848615	5.346331	7.053877	4.510798	4.469209
	Mean	3.1452	9.84567	12.01134	14.62858	7.186367	8.012648
	StD	0.044171	2.066479	2.130831	5.01671	2.296114	3.08245
F5	Best	0	0	1.834959	0.234594	0.914269	2.018337
	Mean	0.005013	0.004231	5.984459	2.675379	2.198233	4.892184
	StD	0	0	0.684503	2.042598	3.142542	1.038934
F6	Best	0	0	0.164169	0.980453	2.472831	1.074761
	Mean	0	0	1.78264	3.176814	2.07127	1.06224
	StD	0.084823	2.08132	1.082067	2.469861	3.183512	2.852861
F7	Best	0.13478	0.923871	0.718115	0.869643	0.596672	0.869383
	Mean	0.216305	2.017324	1.096807	1.742168	2.181644	1.287041
	StD	0.005104	0.191874	1.152126	1.73561	0.109945	1.021639
F8	Best	1.0413068	6.143789	7.48119	6.098491	14.07887	12.50496
	Mean	2.126032	7.039484	10.022987	18.92794	14.08586	9.65414
	StD	0.91876	3.581299	5.341865	4.392864	4.463289	6.96191
F9	Best	0	0	12.20025	0.927324	0.908194	7.624522
	Mean	0	0	21.941308	3.028442	2.896406	17.53869
	StD	0	0	7.091461	3.098481	2.098772	12.72391
F10	Best	0.037001	1.861761	5.912318	4.13942	5.399721	4.896213
	Mean	1.027213	4.94128	10.357041	13.76097	8.951209	6.492021
	StD	0.065305	3.391351	4.180178	6.629183	5.934245	9.058511
F11	Best	0	0	0.447832	1.008874	1.131735	1.051882
	Mean	0.000744	0.097753	0.989018	0.810364	0.32697	0.614351
	StD	0	0	0.091723	0.923161	0.143592	0.148456
F12	Best	0	0	0	0	0	0
	Mean	0	0	0	0	0	0
	StD	0	0	0	0	0	0

Table 6. Comparison of the outcomes gained through the EGTO and the other employed optimizers.

of Table 6 highlights the EGTO algorithm's effectiveness, reliability, and performance in a range of benchmark functions. Its capability to surpass novel metaheuristic algorithms in optimizing diverse problem types demonstrates its potential as a reliable and effective optimization tool for various real-world uses.

There are several other reasons on which EGTO could have surpassed other optimizers in efficacy. The first reason is the balancing between exploitation and exploration; many algorithms cannot strike a balance between them that can result in suboptimal solutions or premature convergence, while EGTO has two improvements like elimination stage and constriction factor; they assist the model in achieving adaptive balance. Those algorithms might represent strong efficiency in diverse optimization problems; however, they usually struggle with highly multimodal or high-dimensional functions. However, the adaptive mechanism of the suggested EGTO enables the algorithm to manage diverse kinds of issues in an efficient manner, which can be supported by the accomplished results.

Simulation results

Using the mentioned examples allows for the derivation of measurement indicators, including accuracy, recall, precision, and F1-score. These indicators work as measures of the model's capability to categorize tweets in an accurate manner. To ensure an unbiased and fair comparison, both the proposed model and other state-of-the-art methods have been evaluated using a 5-fold cross-validation method. This method has been extensively accepted and reliable for assessing a model's efficacy and its ability to generalize. The procedure includes division of the data into 5 random and equal subsets, using all subsets as a test set once, and utilizing the remaining

Model	Accuracy	Precision	Recall	F1-score
CNN-EGTO	98	95	98	96.47
SVM	90	90	92	90.98
KNN	91	89	88	88.49
Naïve Bayes	94	86	90	87.95
LSTM	89	96	92	93.95

Table 7. Classification efficacy of different optimizers for positive polarity.

Model	Accuracy	Precision	Recall	F1-score
CNN-EGTO	96	97	98	97.49
SVM	92	94	90	91.95
KNN	90	87	85	85.98
Naïve Bayes	91	88	87	87.49
LSTM	90	91	90	90.49

Table 8. Classification accuracy of the suggested model and other model in classifying the tweets regarding negative polarity.

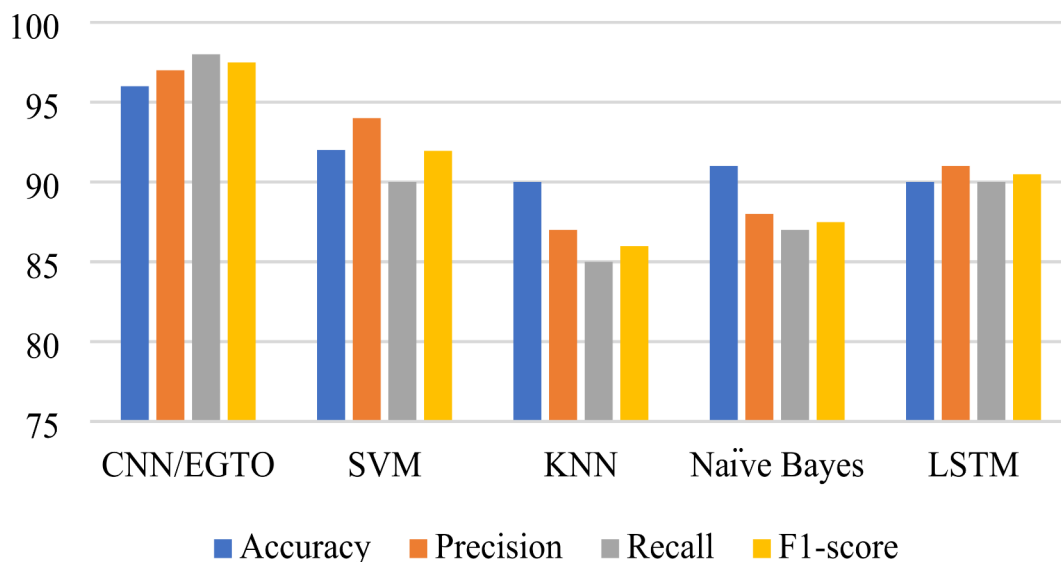


Fig. 3. The performance metrics of various approaches and the suggested in classifying the tweets regarding positive polarity.

9 subsets as a training set. The outcomes of the five folds are then averaged in order to calculate the final performance metrics.

A comparative efficacy assessment of the models has been presented regarding correctly forecasting polarity. The outcomes of accuracy, recall, precision, and F1-score have been gained for the suggested EGTO and five other optimizers, including 4 advanced optimizers, such as SVM, KNN, Naïve Bayes, and LSTM. The Table 7 illustrates all the data relevant to the accuracy, precision, recall, and F1-score achieved by the aforementioned optimizers. Here, Tables 7 and 8 showcase the performance metrics, including accuracy, precision, recall, and F1-score, of the CNN-EGTO model in comparison with other deep learning and machine learning methods for classifying both positive and negative polarities.

It can be observed from the Table 7 that the suggested model could achieve the highest and the most optimum value of accuracy in the present study. In other words, it means that this method could outperform the other optimizers in accurately classifying the tweets of different users regarding the positive polarity.

The comparison results are shown in Fig. 3, which visually depicts the performance measures of all approaches. The CNN/EGTO model performs better than the other approaches across most metrics and accomplishes the maximum average values for Recall, Accuracy, and F1-score. Except, in the case of precision, only CNN could perform better than the suggested model. Additionally, Fig. 3 illustrates the consistency and resilience of the suggested model, with minimal variance and error rates observed in various iterations.

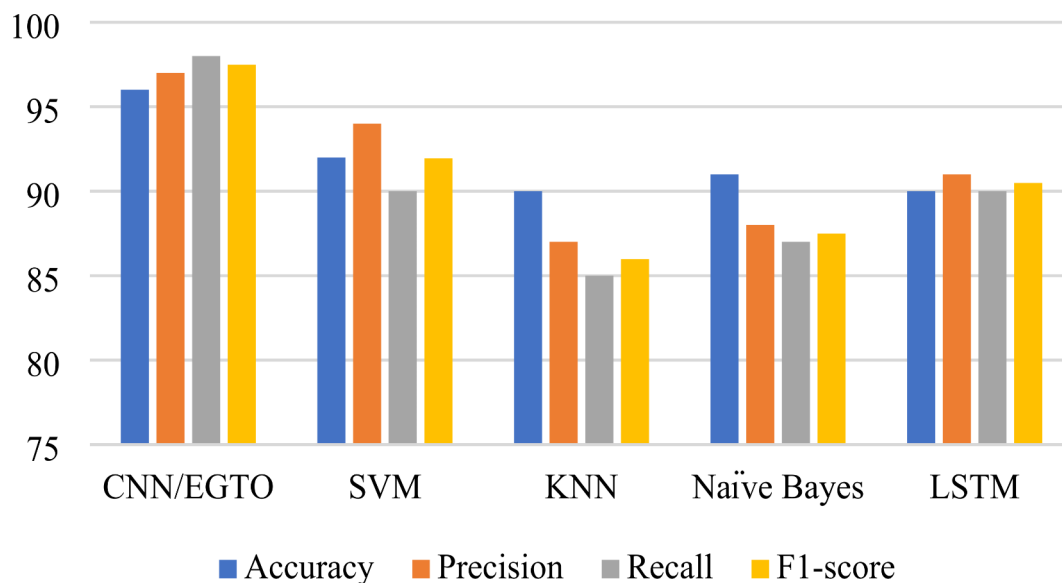


Fig. 4. The classification efficacy of the suggested model in comparison with several other models regarding negative polarity.

The suggested model displays a high level of Precision, accurately classifying a considerable portion of instances to belong to a particular group. The Precision value of the suggested network is 95%, surpassing that of other methods like SVM, KNN, and Naïve Bayes. However, the LSTM could outperform the suggested model by 1%. Furthermore, the suggested model showcases a great level of Accuracy that can classify a considerable portion of all instances in accurate manner. The Accuracy of the suggested network is 98%, outperforming other networks like KNN (91%), SVM (90%), and Naïve Bayes (94%). This represents the model's ability to classify positive tweets with high precision and recall that could minimize false negatives and positives.

Additionally, the suggested network demonstrates a great level of Recall, accurately categorizing a considerable portion of instances that truly belong to a particular group. The Recall value of the suggested network is 98%, surpassing that of other approaches. Moreover, the suggested model could achieve F1-score value of 96.47%, indicating a great balance between recall (98%) and precision (95%); it represents the model's whole robustness in recognizing true positive samples efficiently.

The outcomes show that the recommended CNN/EGTO network constantly acts well, demonstrating its impressive stability and resilience. In comparison with other methods, the suggested model demonstrates minimal error rates and variation when used on various iterations. Furthermore, the findings indicate the effective efficacy of the suggested model in a range of tweet categories. The current model efficiently captures essential patterns and features from tweets, allowing accurate categorization into the proper groups.

Previously, the classification performance of the suggested model and several other networks were assessed. It was revealed that the suggested model could outperform other models successfully regarding positive polarity. Furthermore, the classification performance of various networks, including SVM, KNN, Naïve Bayes, LSTM, and the suggested CNN-ETGOA, must be evaluated when dealing with negative polarity. The Table 8 fully illustrates the efficiency of the networks regarding various efficacy metrics, such as recall, precision, F1-score, and accuracy.

The table presented above has fully illustrated the recommended model could accomplish the best value for accuracy in this research. Moreover, this can be interpreted in a way that the suggested approach was able to perform superior to the several other algorithm regarding classifying the tweets carrying negative polarity.

In the following, the findings have been represented in Fig. 4, depicting the performance metric of the previously mentioned models. It is crystal clear that the recommended model was able to perform superior to the other network and accomplished the highest values in terms of recall, accuracy, precision, and F1-score. In addition, Fig. 4 can fully demonstrate the resilience and consistency of the recommended model that had really minor variance regarding diverse iterations.

It can be indicated via the Fig. 4 that the model could achieve really high values for Precision, which means that the model could classify significant portion of instance in an accurate manner, belonging to a specific class. It is worth noting that the recommended model could accomplish the precision value of 97%, outperforming other networks like KNN (87%), SVM (94%), and Naïve Bayes (88%). This represents the model's capacity to categorize positive tweets with high precision. Also, the recommended network indicated a great accuracy level that could categorize substantial amount of instance accurately. The CNN/EGTO could gain the accuracy level of 96%, which could exceed other optimizers. In addition, an incredible recall value was obtained, hence categorizing substantial amount of instances that belonged to a specific group. The recommended model could gain the recall value 98%. Eventually, it should be mentioned that the recommended network achieved the value of 97.49% for F1-score.

The outcome displayed that the CNN-EGTO could perform well in constant way. It displayed that this model could have notable resilience and consistency. The suggested model depicted minor variation and rates of error compared to other models, when diverse iterations were run. Besides,

The findings support the efficacy of the suggested model as a powerful deep learning structure to address classification of tweets. By combining a CNN with the Enhanced Gorilla Troops Optimization Algorithm (EGTO) algorithm, the suggested network optimized the parameters and weights of the network. As a result, the suggested model demonstrated outstanding efficiency, dependability, and accuracy in the classification of tweets regarding negative and positive polarities, outperforming other cutting-edge approaches. Furthermore, the suggested model showed potential in dealing with real-world categorization challenges that required great levels of efficiency, dependability, and accuracy.

Discussing practical implications, it can be said that ensuring high precision in identifying positive feedback guarantees that customer responses marked as positive are true, which minimizes risks in practical scenarios such as targeted marketing efforts. Likewise, achieving high recall in identifying negative feedback reduces the chances of failing to detect negative sentiments, which is essential for areas like managing customer complaints or monitoring social channels to prevent crises.

Discussion

The findings of this research indicate that the CNN-EGTO model significantly surpasses conventional sentiment analysis models regarding recall, precision, accuracy, and F1-score. Attaining an accuracy of about 98% for tweets with positive sentiment shows that the suggested model effectively captures the complex emotions conveyed in social media texts. This level of performance is especially impressive considering the inherent difficulties in analyzing tweets, such as contextual ambiguities, shorthand, and informal language.

In comparison with earlier research in this area, the results are consistent with those of Umer et al.²⁵, who combined Long Short-Term Memory (LSTM) with CNN to improve the accuracy of sentiment predictions, although their method resulted in greater complexity and longer training durations. Additionally, the model proposed in this article achieves a balance between exploitation and exploration by utilizing the EGTO for hyperparameter optimization. This indicates that the suggested approach could be a more feasible option for applications in real-time sentiment analysis.

Furthermore, the findings differ from those of Gandhi et al.²⁷, who observed decreased accuracy levels because of insufficient preprocessing methods. The thorough preprocessing steps were implemented, which involved managing noise and correcting spelling mistakes, that could play a significant role in the enhanced efficacy of the CNN-EGTO model. This underscores the essential role of data preparation and preprocessing stages in machine learning applications, especially in the field of natural language processing.

The preprocessing techniques utilized in this study were essential for achieving elevated accuracy rates. By overcoming issues such as correcting spelling errors and noise elimination, the quality of the input data was significantly improved. Therefore, it can be concluded that developing standardized preprocessing protocols for the analysis of social media text is highly necessary.

Conclusion

In recent years, CNN has become a widely used technique for sentiment analysis. In this investigation as well, a dataset was generated, comprising tweets with different polarities - positive, negative, and neutral, for sentiment analysis, in order to introduce a novel hybrid approach based on CNN optimized by Enhanced Gorilla Optimization Algorithm (EGTO). For this purpose, tweets were gathered from Twitter to establish a large dataset and propose sentiment analysis using a convolutional neural network that is optimized by EGTO. The supervised learning approach of sentiment analysis was employed in the present study to classify the tweets. The suggested method aimed to categorize people's Twitter posts into three primary classes, including positive, negative, and neutral. As a result, it has now become feasible to draw important inferences about individuals' feelings. There were some measurement metrics employed in the present, including precision, accuracy, recall, and F1-score. The model's performance was assessed and compared against other cutting-edge models for sentiment analysis. Furthermore, the testing process also involved the use of LSTM, SVM, Naive Bayes, and KNN methods, and the CNN-EGTO achieved the best accuracy rate of classification. Lastly, the proposed approach's classification efficacy was compared to several methods, demonstrating that it exhibited the best performance. In summary, the current study introduced an exceptionally successful technique with accuracy, precision, recall, and F1-score values of 98%, 95%, 98%, and 96.47%, respectively, for positive polarity. Furthermore, the suggested model could gain the values of 97%, 96%, 98%, and 97.49% for precision, recall, accuracy, and F1-score, respectively. In the end, there are some recommendations for future directions. Although this model could achieve high performance and efficiency, several other sources from diverse platforms of social media, like Instagram and Facebook, for a broader comprehension sentiment in various contexts. Moreover, other datasets with some other language can be good options for scholars to work on. Since English is an international language, it has been worked a lot; however, some other languages, like Spanish, German, and Persian, can be good options. The dependence of particular datasets also, like SemEval-2016, might limit the generalizability of the outcomes. In addition, the quickly alteration of social media can pose some problems and might need other preprocessing stages. Moreover, recognizing ironic or sarcastic contexts is still a problem in this area, since they depend on context, which makes it really hard for the model to detect polarity of those sentences.

Data availability

All data generated or analysed during this study are included in this published article.

Received: 14 October 2024; Accepted: 2 January 2025

Published online: 04 January 2025

References

1. Acheampong, F. A., Wenyu, C. & Nunoo-Mensah, H. Text-based emotion detection: advances, challenges, and opportunities. *Eng. Rep.* **2**, e12189 (2020).
2. Mehta, P., Pandya, S. & Kotecha, K. Harvesting social media sentiment analysis to enhance stock market prediction using deep learning. *PeerJ Comput. Sci.* **7**, e476 (2021).
3. Lo, S. L., Chiong, R. & Cornforth, D. Ranking of high-value social audiences on Twitter. *Decis. Support Syst.* **85**, 34–48 (2016).
4. Bello, A., Ng, S. C. & Leung, M. F. A BERT framework to sentiment analysis of tweets. *Sensors* **23**, 506 (2023).
5. He, B., Macdonald, C., He, J. & Ounis, I. in *Proceedings of the 17th ACM conference on Information and knowledge management*. 1063–1072.
6. Melville, P., Gryc, W. & Lawrence, R. D. in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1275–1284.
7. Balahur, A. et al. Sentiment analysis in the news. *arXiv preprint arXiv:1309.6202* (2013).
8. Turney, P. D. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. *arXiv preprint cs/0212032* (2002).
9. Pang, B., Lee, L. & Vaithyanathan, S. Thumbs up? Sentiment classification using machine learning techniques. *arXiv Preprint cs/0205070* (2002).
10. Hu, M. & Liu, B. in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 168–177.
11. Mir, A. A. & Sevukan, R. Sentiment analysis of Indian tweets about Covid-19 vaccines. *J. Inform. Sci.* **50**, 1308–1320 (2024).
12. Yang, S., Xing, L., Li, Y. & Chang, Z. Implicit sentiment analysis based on graph attention neural network. *Eng. Rep.* **4**, e12452 (2022).
13. Jahin, M. A., Shovon, M. S. H., Mridha, M., Islam, M. R. & Watanobe, Y. A hybrid transformer and attention based recurrent neural network for robust and interpretable sentiment analysis of tweets. *Sci. Rep.* **14**, 24882 (2024).
14. Kumar, S., Singh, R., Khan, M. Z. & Noorwali, A. Design of adaptive ensemble classifier for online sentiment analysis and opinion mining. *PeerJ Comput. Sci.* **7**, e660 (2021).
15. Singh, L. G. & Singh, S. R. Sentiment analysis of tweets using text and graph multi-views learning. *Knowl. Inf. Syst.*, 1–21 (2024).
16. Ritter, A., Clark, S. & Etzioni, O. in *Proceedings of the 2011 conference on empirical methods in natural language processing*. 1524–1534.
17. Althoff, T., Clark, K. & Leskovec, J. Large-scale analysis of counseling conversations: an application of natural language processing to mental health. *Trans. Association Comput. Linguistics.* **4**, 463–476 (2016).
18. Calvo, R. A., Milne, D. N., Hussain, M. S. & Christensen, H. Natural language processing in mental health applications using non-clinical texts. *Nat. Lang. Eng.* **23**, 649–685 (2017).
19. Larsen, M. E. et al. We feel: mapping emotion on Twitter. *IEEE J. Biomedical Health Inf.* **19**, 1246–1252 (2015).
20. Dini, L. & Bittar, A. in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. 3953–3958.
21. Hausteijn, S. et al. Tweets as impact indicators: examining the implications of automated bot accounts on T. twitter. *J. Association Inform. Sci. Technol.* **67**, 232–238 (2016).
22. Paltoglou, G. & Giachanou, A. Opinion retrieval: Searching for opinions in social media. *Professional Search in the Modern World: COST Action IC1002 on Multilingual and Multifaceted Interactive Information Access*, 193–214 (2014).
23. Muhammad, A., Wiratunga, N. & Lothian, R. in *2014 IEEE 26th International Conference on Tools with Artificial Intelligence*. 461–468 (IEEE).
24. Jianqiang, Z., Xiaolin, G. & Xuejun, Z. Deep convolution neural networks for twitter sentiment analysis. *IEEE Access.* **6**, 23253–23260 (2018).
25. Umer, M. et al. Sentiment analysis of tweets using a unified convolutional neural network-long short-term memory network model. *Comput. Intell.* **37**, 409–434 (2021).
26. Tam, S., Said, R. B. & Tanriöver, Ö. Ö. A ConvBiLSTM deep learning model-based approach for Twitter sentiment classification. *IEEE Access.* **9**, 41283–41293 (2021).
27. Gandhi, U. D., Kumar, M., Chandra Babu, P. & Karthick, G. G. Sentiment analysis on twitter data by using convolutional neural network (CNN) and long short term memory (LSTM). *Wireless Pers. Commun.*, 1–10 (2021).
28. Meena, G., Mohbey, K. K., Acharya, M. & Lokesh, K. Original Research Article An improved convolutional neural network-based model for detecting brain tumors from augmented MRI images. *J. Auton. Intell.* **6** (2023).
29. Meena, G., Indian, A., Mohbey, K. K. & Jangid, K. Point of interest recommendation system using sentiment analysis. *J. Inform. Sci. Theory Pract.* **12**, 64–78 (2024).
30. Meena, G., Mohbey, K. K. & Kumar, S. Monkeypox recognition and prediction from visuals using deep transfer learning-based neural networks. *Multimedia Tools Appl.*, 1–25 (2024).
31. Hangya, V. & Farkas, R. A comparative empirical study on social media sentiment analysis over various genres and languages. *Artif. Intell. Rev.* **47**, 485–505 (2017).
32. Lima, A. C. E. De Castro, L. N. A multi-label, semi-supervised classification approach applied to personality prediction in social media. *Neural Netw.* **58**, 122–130 (2014).
33. Sedhai, S. & Sun, A. in *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 223–232.
34. Chikersal, P., Poria, S. & Cambria, E. in *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*. 647–651.
35. Saberi, Z. A., Sadr, H. & Yamaghani, M. R. in *10th International Conference on Artificial Intelligence and Robotics (QICAR)*. 131–137 (IEEE). (2024).
36. Jadidinejad, A. H. & Sadr, H. Improving weak queries using local cluster analysis as a preliminary framework. *Indian J. Sci. Technol.* **8**, 495–510 (2015).
37. Sadr, H., Soleimandarabi, M. N., Pedram, M. & Teshnelab, M. in *5th International Conference on Web Research (ICWR)*. 134–140 (IEEE). (2019).
38. Sadr, H., Salari, A., Ashoobi, M. T. & Nazari, M. Cardiovascular disease diagnosis: a holistic approach using the integration of machine learning and deep learning models. *Eur. J. Med. Res.* **29**, 455 (2024).
39. Malhotra, R. & Singh, P. Recent advances in deep learning models: a systematic literature review. *Multimedia Tools Appl.* **82**, 44977–45060 (2023).
40. Li, S. et al. Evaluating the efficiency of CCHP systems in Xinjiang Uygur Autonomous Region: an optimal strategy based on improved mother optimization algorithm. *Case Stud. Therm. Eng.* **54**, 104005 (2024).
41. Cai, W. et al. Optimal bidding and offering strategies of compressed air energy storage: a hybrid robust-stochastic approach. *Renew. Energy.* **143**, 1–8 (2019).

42. Chang, L., Wu, Z. & Ghadimi, N. A new biomass-based hybrid energy system integrated with a flue gas condensation process and energy storage option: an effort to mitigate environmental hazards. *Process Saf. Environ. Prot.* **177**, 959–975 (2023).
43. Cao, Y. et al. Optimal operation of CCHP and renewable generation-based energy hub considering environmental perspective: an epsilon constraint and fuzzy methods. *Sustainable Energy Grids Networks*. **20**, 100274 (2019).
44. Han, E. & Ghadimi, N. Model identification of proton-exchange membrane fuel cells based on a hybrid convolutional neural network and extreme learning machine optimized by improved honey badger algorithm. *Sustain. Energy Technol. Assess.* **52**, 102005 (2022).
45. Alferaidi, A. et al. Distributed Deep CNN-LSTM Model for Intrusion Detection Method in IoT-Based Vehicles. *Mathematical Problems in Engineering* (2022). (2022).
46. Zhu, L. et al. Multi-criteria evaluation and optimization of a novel thermodynamic cycle based on a wind farm, Kalina cycle and storage system: an effort to improve efficiency and sustainability. *Sustainable Cities Soc.* **96**, 104718 (2023).
47. Umair, M. B. et al. A Network Intrusion Detection System Using Hybrid Multilayer Deep Learning Model. *Big data* (2022).
48. Ghiasi, M., Ghadimi, N. & Ahmadiania, E. An analytical methodology for reliability assessment and failure analysis in distributed power system. *SN Appl. Sci.* **1**, 1–9 (2019).
49. Guo, H., Gu, W., Khayatnezhad, M. & Ghadimi, N. Parameter extraction of the SOFC mathematical model based on fractional order version of dragonfly algorithm. *Int. J. Hydrog. Energy*. **47**, 24059–24068 (2022).
50. Karamnejadi Azar, K. et al. Developed design of battle royale optimizer for the optimum identification of solid oxide fuel cell. *Sustainability* **14**, 9882 (2022).
51. Clerc, M. & Kennedy, J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.* **6**, 58–73 (2002).
52. Mohamed, A. W., Hadi, A. A. & Mohamed, A. K. Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. *Int. J. Mach. Learn. Cybernet.* **11**, 1501–1529 (2020).
53. Zhao, W., Wang, L. & Zhang, Z. Artificial ecosystem-based optimization: a novel nature-inspired meta-heuristic algorithm. *Neural Comput. Appl.* **32**, 9383–9425 (2020).
54. Trojovský, P. & Dehghani, M. Pelican optimization algorithm: a novel nature-inspired algorithm for engineering applications. *Sensors* **22**, 855 (2022).
55. Hashim, F. A. & Hussien, A. G. Snake Optimizer: a novel meta-heuristic optimization algorithm. *Knowl. Based Syst.* **242**, 108320 (2022).
56. Heidari, A. A. et al. Harris hawks optimization: Algorithm and applications. *Future Generation Comput. Syst.* **97**, 849–872 (2019).

Author contributions

Fang Li, Jialing Li and Francis Abza wrote the main manuscript text and prepared figures. All authors reviewed the manuscript.

Funding

This study was funded by the Education and Teaching Reform Research Project of Chongqing College of International Business and Economics (JG2023021).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.L. or F.A.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025