Analyzing Sentiments using Optimized Novel Ensemble Fuzzy and DL based Approach with Efficient Feature Selection and Extraction Models

Potu Bharath
School of Computer Science Engineering
Vellore Institute of Technology, India
bharath.21phd7158@vitap.ac.in

Dasari Venkata Lakshmi
School of Computer Science Engineering
Vellore Institute of Technology, India
venkatalakshmi.d@vitap.ac.in

Abstract: *The most popular and active area of data mining study is sentiment analysis. Twitter is a crucial platform for collecting and distributing people's thoughts, feelings, views, and attitudes regarding specific entities. There are several social media networks available today. In light of this, sentiment analysis in the Natural Language Processing (NLP) field became fascinating. Various methods for analyzing sentiment have been explored. However, improvements are still required regarding reliability and system efficacy. Additionally, user emotional expressions typically take the form of naturally occurring human-written textual data with numerous noises and ambiguities. The intricate contextual significance of sentiment expressions is difficult for present studies on sentiment analysis to precisely capture and interpret, particularly in linguistics with complex frameworks. To address these issues, we presented a new integrated fuzzy neural network. The proposed framework is developed for effective and efficient feature selection and hybrid approach based sentiment analysis. Ensemble of novel Deep Convolutional Neuro-Fuzzy Inference System (DCNFIS) and Deep Learning-based (DL) Long short-term memory neural network multilayer stacked bidirectional LSTM neural network analyzes the sentiments. The provided dataset is initially cleaned up and filtered out as part of preprocessing. Utilizing the preprocessed data, sentiment-based features are extracted using the inception-ResNet-V2 model. Then, the relevant features are selected by employing the Enhanced Reptile Search Algorithm (ERSA). The Al-Biruni Earth Radius (BER) optimization algorithm is used to optimize the hyperparameters of the ensemble approach, which analyzes the sentiment categories such as positive, negative, very positive, very negative, and neutral. Finally, an effectiveness assessment of the suggested and present classifiers is presented. Using three distinct research datasets, we conducted an experimental evaluation of the suggested model. While differentiating from the proposed approach, the proposed approach yields a greater performance of 98.97% recall, 99.06% precision, 99.13% accuracy and 99.01% F1-score. The experimental investigation analyzes that the proposed approach gains superior performance over existing approaches*.

Keywords: *Sentiment analysis, deep convolutional fuzzy neural network, multilayer stacked bidirectional LSTM neural network inception-ResNet-V2, enhanced reptile search algorithm, Al-Biruni earth radius optimization algorithm*.

Received February 12, 2024; accepted July 6, 2024
https://doi.org/10.34028/iajit/21/4/14

1. Introduction

The growth of content created by users in online communities and websites such as Trip Advisor, Amazon, Facebook, Twitter, and Instagram has resulted in social media being the ultimate venue for expressing thoughts about events, products, and services [8]. This ability, combined with the rapid diffusion of online content, has raised the worth of the viewpoints submitted. Various activities involving Natural Language Processing (NLP) are currently being used to analyze this massive volume of data [25]. Emotional analysis is a significant analysis now used to classify sentiments and opinions created by humans and in text. In sentiment analysis, machine learning methods are commonly utilized. At first, the Bag of Words (BoW) algorithm was modified, which maps a specific feature vector to a page and then divides it using machine learning algorithms [18, 22, 27]. This system was praised for its effectiveness and simplicity. The disadvantage of this procedure entails that the initial language itself is lost, synthesis patterns are broken, and word structure is lost. As a result, novel approaches such as higher-order n-grams were developed [6].

The goal of feelings investigation is to evaluate and derive data gathered from subjective material presented on websites. Analyzing sentiment has lately been a popular study in information extraction and NLP because of various educational and corporate programs and the rapid growth of Web 2.0 [15, 16, 17, 23]. As a result, several strategies and instruments for defining a document's orientation have recently been created. Polarization identification is a binary categorization task used in many sentiment analysis applications. For achieving excellent valence categorization results, most previous approaches for sentiment measurement trained simple models with relevant features [4, 7]. On linguistic variables like n-grams, Part-Of-Speech (POS) tags, and semantic characteristics, these models often used standard categorization approaches such as Support Vector Machines (SVM), Linear Discriminant

Analysis (LDA), and Naive Bayes (NB) [5]. This technique has two significant disadvantages:

1) The feature set that determines how the Algorithm will be educated is dense and high-dimensional, causing decreased system performance.
2) The characteristic engineering procedure is labour and time-intensive [1, 9].

Scalability and parameter tuning may be challenges for SVM, and NB is less accurate in complex datasets due to its assumption of feature independence. Similarly, BoW streamlines text representation but might miss subtleties in semantics. The introduction could make a stronger case for the necessity and superiority of the suggested approach in improving sentiment analysis outcomes by drawing a comparison between these limitations and the novel aspects of the suggested strategy perhaps highlighting developments in the extraction of features, Deep Learning (DL) integration, or performing of noisy data.

Deep neural networks have been proposed for textual data analysis, with the primary goal of acquiring word integration or conducting tasks involving machine learning like categorization and grouping on the learned feature vectors [10, 26]. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) were the most commonly used deep network kinds in text processing research. The capacity of CNNs to learn local trends and the efficacy of RNNs in consecutive modeling are the reasons for their appeal [11]. When the input data contains long-term dependencies, RNNs despite becoming beneficial for several analyses of text programs, experience vanishing and inflating gradients [14, 21]. These interactions are prevalent in most NLP uses, especially those that analyze sentiment.

1.1. Motivation

Sentiment analysis involves identifying and categorizing the opinions or polarity of textual data, which is especially relevant given the vast amount of information generated on social media platforms like Twitter. Analyzing online sentiment can provide valuable insights into public opinion on various products and topics. However, Twitter Sentiment Analysis (TSA) presents several challenges, primarily due to the character limit of tweets and the frequent use of emoticons, slang, and misspellings, which necessitate an extra preprocessing stage to clean the raw data.

Current approaches in TSA, particularly those based on DL, have not achieved meaningful results due to slower manual text labelling, the use of smaller datasets, and the presence of extraneous information and gaps that diminish classification effectiveness. Additionally, the dimension of features extracted in existing methods further impacts the efficiency of DL models.

The goal of this research is to develop a novel feature extraction and selection method integrated with DL models to address these challenges. The proposed approach aims to improve the efficiency of sentiment classification by enhancing preprocessing steps to eliminate unnecessary details from the dataset, thereby reducing the processing time for feature extraction.

Preprocessing explicitly addresses TSA issues in the following ways:

- Character limit and conciseness: advanced tokenization handle the short length of tweets, ensuring meaningful information is retained and irrelevant data is discarded.
- Handling slang, emoticons, and misspellings: specific steps for normalizing slang, converting emoticons to text representations, and correcting common misspellings ensure accurate sentiment interpretation.
- Eliminating extraneous information: stop word removal and stemming reduce the dataset to its most essential components, focusing the analysis on words that carry significant sentiment weight.
- Feature dimensionality reduction: preprocessing eliminates redundant and irrelevant information, reducing the complexity of feature dimensions and streamlining the data for more effective feature extraction and selection.

An optimization-based feature selection process is implemented to focus on relevant features, improving the overall analysis. The suggested method stands out by effectively evaluating text-based characteristics compared to current algorithms. By incorporating fuzzy mechanisms with DL, the proposed algorithm simplifies the model and enhances its ability to learn selected features, leading to better performance. The new ensemble approach, Deep Convolutional Neuro-Fuzzy Inference System (DCNFIS), utilizes DL to automatically identify complex correlations and patterns in the data, addressing the limitations of existing methods and improving the effectiveness of sentiment analysis.

1.2. Novel Contribution

The primary essential contributions of this research are as follows:

- To analyze the sentiment from tweets, we introduced the ensemble novel approach of DCNFIS and multilayer stacked bidirectional LSTM neural network. The hyperparameters are optimized by employing the Al-Biruni Earth Radius (BER) Optimization Algorithm to enhance the performance even more.
- The essential features are extracted with the help of the inception-ResNet-V2 approach. With smaller feature dimensions, training and prediction times are speed up.
- The Enhanced Reptile Search Algorithm (ERSA) selects the relevant features. Less features make the

model more computationally tractable and memory-efficient.
- We utilized three benchmark datasets and various metrics to compare the proposed performance with conventional DL and machine learning techniques.

The following summarizes the paper's structure: part 2 explains the prior studies, part 3 provides a detailed explanation of our suggested approach, part 4 covers the setup of the experiment and part 5 provides a summary of the research.

2. Related Work

The expansion of user-generated material on the internet has created a massive source of valuable sentiment data, increasing interest in SA. The prior research is discussed in this section.

An Attention-based Bidirectional CNN-RNN Deep Model (ABCDM) by Basiri *et al*. [2]. ABCDM will retrieve past and future situations by considering the temporal movement of data in each direction using two independent layers. Additionally, the mechanism for attention is used on the bidirectional ABCDM layer results to emphasize specific phrases more or less than others. ABCDM uses convolution and pooling algorithms to decrease the dimensionality of the characteristics and retrieve position-invariant local information. The most typical and crucial activity of sentiment analysis, sentiment polarity identification, is used to assess the performance of ABCDM.

To examine both the sentiment analysis results from the first stage and the technical stock market indicators, Jing *et al*. [13] suggested a combination study framework that employed the LSTM neural network technique. In addition, this research has conducted real-world trials on the Shanghai Stock Exchange (SSE) in six key industries over three time periods to validate the effectiveness and applicability of the proposed model. The results of the experiment demonstrate that the proposed method outperforms the initial predictive algorithms in investor sentiment classification.

A technique using DL to analyze the sentiment of consumer feedback from Twitter is presented by Onan *et al*. [24]. In the structure being discussed, CNN-LSTM and TF-IDF weighted Glove word embedding are integrated. Five layers make up the recommended framework. The performance of weighted word embedding methods was superior to that of unweighted approaches. Furthermore, combining CNN with the LSTM framework can produce encouraging outcomes.

Behera *et al*. [3] hybrid strategy of CNN-LSTM deep learning frameworks is recommended for the sentiment categorization of evaluations posted at various domains. While neural networks frequently produce positive outcomes in the sequential assessment of a prolonged text, deep convolutional networks have succeeded exceptionally in the local feature selection process. Two analyses of sentiment goals are the main focus of the proposed Co-LSTM model. According to the experimental findings, the suggested ensemble model performs better than the existing machine learning methods regarding precision and other factors.

Using Robustly Optimized BERT pre-Training Approach (RoBERTa), Liao *et al*. [19] offer a multi-task aspect-category sentiment analysis model. To extract features from text and aspect tokens, they used the RoBERTa based on a deep bidirectional Transformer, considering every aspect grouping as a separate job. We then use the cross-attention method to enable the model to concentrate on the most pertinent features of the specific aspect category. The suggested model outperformed various other approaches for evaluation in aspect-category sentiment analysis, depending on the experimental data. The impacting aspects of our suggested model are also examined.

In order to create embeddings of words, Jain *et al*. [12] introduce a Dilated Convolutional Neural Network (BERT-DCNN) framework based on Bidirectional Encoder Representations from Transformers (BERT). BERT was utilized as a pre-trained language framework. Further aiding in the algorithm's tuning are three separate layers of a DCNN stacked with a global average pooling layer.

2.1. Research Gap

Despite the availability of massive datasets, several challenges persist in achieving effective accuracy, processing time, and complexity in sentiment analysis. Specific difficulties encountered in earlier studies include issues with dataset size, processing speed, feature extraction, and classifier performance. Large datasets pose significant challenges in terms of processing speed and computational resources required for analysis, making it difficult to handle and analyze the data efficiently within reasonable timeframes. This inefficiency not only delays obtaining results but also increases the complexity of the analysis, hindering effective sentiment classification.

Moreover, extracting relevant and high-level features is crucial for accurate sentiment analysis. Many existing approaches, however, struggle with the extraction of irrelevant and low-level characteristics, which diminishes the effectiveness of classifiers. These irrelevant features not only occupy substantial storage space but also introduce unnecessary complexity, leading to higher misclassification rates and reduced overall accuracy of sentiment analysis models.

The proposed architecture aims to address these critical issues by introducing a more efficient method for feature extraction and selection. By focusing on extracting and utilizing only the most relevant features, the algorithm aims to streamline processing speed and reduce computational complexity. This optimized approach not only enhances feature extraction through advanced techniques but also improves classifier performance by excluding irrelevant features. Consequently, the proposed method is expected to achieve lower misclassification error rates and higher accuracy across multiple datasets, effectively overcoming the inefficiencies present in current sentiment analysis methodologies. Table 1 illustrates the summary of related works.

Table 1. Summary of related studies.

Reference	Method used	Merit	Demerit
Basiri *et al*. [2]	ABCDM	Effective in managing data in sequence	Long-term dependencies might not be adequately captured
Jing *et al*. [13]	LSTM	Efficient for memory retention and sequential data	susceptible to the issue of vanishing gradients
Onan *et al*. [24]	CNN-LSTM	Captures spatial and temporal features	Overfitting risk due to complex architecture
Behera *et al*. [3]	CNN-LSTM	combines learning in two phases: spatial and sequential	requires precise adjustment of the hyperparameters
Liao *et al*. [19]	RoBERTa	State-of-the-art performance in natural language tasks	High computational cost, large memory requirement
Jain *et al*. [12]	BERT-DCNN	Utilizes contextual embeddings, strong performance	Limited interpretability of deep neural network models

3. Proposed Methodology

The analysis of sentiments, often known as opinion mining, is an NLP activity that determines the emotional undertone or sentiment present in written content, such as feedback, social media posts, or opinions. The present research suggests a new deep fuzzy model to solve the shortcomings of the available deep frameworks for analyzing sentiment. The next step after data collection is tweet preprocessing. Tokenization, stop word removal, stemming, and the elimination of @ and wrong symbols are all done in the preprocessing phase.

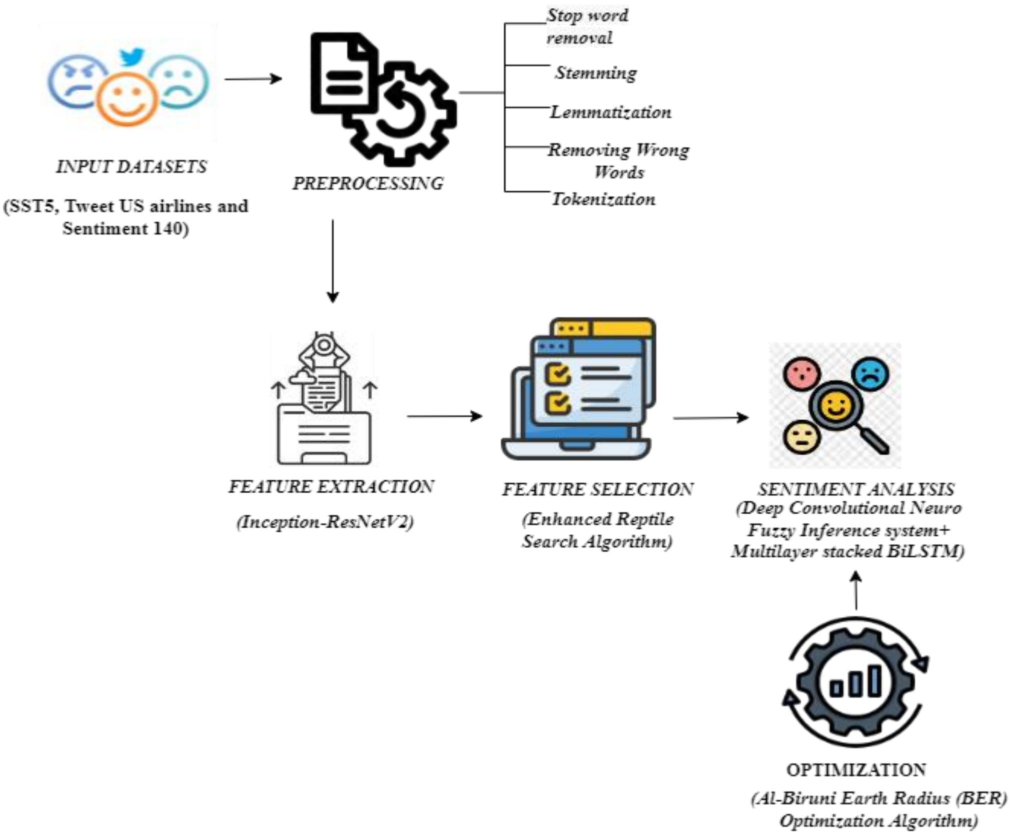


Figure 1. Proposed architecture diagram.

The preprocessed dataset has been utilized as input for the following procedure. The features are extracted using inception-ResNet-V2 as a base [20]. The Improved Reptile Search (IRS) method is then employed to choose the best features. The output of IRSA is then used to analyze sentiment using five distinct types using the ensemble of DCNFIS and multilayer stacked bidirectional LSTM neural network. The BER optimization algorithm optimizes the hyperparameters from DCNFIS. A graphical illustration of the suggested approach is shown in Figure 1.

3.1. Preprocessing

Preprocessing is a technique that helps organize the dataset by carrying out fundamental operations before providing it to a model, such as removing blank spaces and invalid words, translating diverse word forms into their core words, removing duplicate words, etc. It transforms the unorganized, raw dataset into a usable dataset for subsequent use.

- **Tokenization**

Continued text is divided into words, symbols, and elements to do this. It should be accurate and effective because it significantly affects how well the subsequent evaluation performs.

- **Stemming**

Stemming removes nouns with inflectional affixes, like playing-play and studies-study. English and Spanish are the two languages that stemming is most effective.

- **Removing @ and Wrong Symbols**

Words beginning with @ are eliminated after stemming because Twitter gives every user an individual name that begins with @. Special characters are then deleted after that. This investigation discovered that a few symbols were still present in the Tweets after the unique symbol-removal step was finished. The removal of symbols (such as a heart) then proceeds to the bad symbol phase.

- **Lemmatization**

When lemmatizing, the morphological examination of the words is considered. With the root words that are part of the sentences, it properly lowers inflected terms. It is also known as a lemma and consists of words in canonical, citation, and dictionary forms. Subsequently, the tweets were tokenized and lemmatized to specific words. The tweets dataset yielded a total of 243297 unique tokens, while the dataset yielded 14255 unique tokens.

- **Stop Word Removal**

Stop words are eliminated from the Tweets in the following phase. Stop words improve the readability of sentences, but they offer no more information for text analysis. The categorization method works more effectively when stop words are eliminated.

3.2. Feature Extraction

In sentiment analysis, feature extraction is essential for preparing and translating unstructured text data into a format that DL models can use. It aids in enhancing the model's performance, decreasing overfitting, and improving the effectiveness and interpretability of the research. Inception-ResNet-V2 is the best choice for extracting features in sentiment analysis because of its depth, effectiveness, and high accuracy. It can substantially aid activities involving sentiment analysis due to its capacity to extract rich, hierarchical characteristics from data.

3.2.1. Inception-Resnet-V2

According to the network framework of inception, employing multiple kernels for convolution of various sizes can increase the network's flexibility and retrieve more abundant information on various dimensions. In addition, using the NIN approach, its inception network framework may significantly decrease its parameters, enabling it to use fewer convolution kernels without compromising the ability to express the model's features. This reduces the model's complexity.

The network training and optimization of parameters are speed up by direct transmitting signals from various units and layers, both backward and forward. The deep coupled ResNet comprises two branch networks and a backbone network. Due to the possibility of differences in feature count between the map of characteristics and the features, 1x1 convolution must be used to boost or lower the size of the convolution residual network. At this point, the operation is represented as:

$$F(x_l) = w * x_l + \alpha \tag{1}$$

$$y_l = R(F) + h(x_l) \tag{2}$$

$$x_{l+1} = R(y_l) \tag{3}$$

These calculations have the following components: input is represented as x, weight is indicated as w, two branches added together is indicated as y_l, the Function of the Relu is denoted as $RF(x_l)$, represents convolution operation, the input straightforward transformation is represented as $h(x_l)$ and x_{l+1} is the residual module's outcome. According to Equation (4), Relu is an activation function that aids in the spread of the ladder.

$$R(x) = max(0, x) \tag{4}$$

While x is more significant than zero, $R(x)=(x)$, and its lead is 1. When x is less than zero, $R(x)=(0)$, with a lead of 0. The integer of x and the threshold value of 0 can be entered into the forward computations to get the result. The gradient in an inverted computation is a value of 1 or 0. In other words, there is barely any gradient drop. Relu is better for deepening networks since it is simpler to estimate and has a shorter gradient decline than Tanh and the Sigmoid.

A residual learning network unit was added to prevent the gradient from vanishing during inception network framework training. In addition, when the network model's effectiveness achieves a specific saturation, the residual network's layer can be transferred similarly, speeding up and facilitating the training network's convergence. The residual function is represented by $F(\bullet)$, while the inputs of the i^{th} and nth units are represented by Xi and Xn, respectively. Therefore, the gradient for learning characteristics across the shallow i layer through the deep n layer in the expression below will never approach zero, regardless of the extent to which the network's layers are layered.

$$\frac{\partial Xn}{\partial Xi} = \frac{\partial Xi + F(Xi, \omega i, \alpha i)}{\partial Xi} = 1 + \frac{\partial F(Xn, \omega n, \alpha n)}{\partial Xn} \tag{5}$$

The remaining network learning units with three layers were employed in our study. Convolutions of 1×1 and 3×3 reduce the structure dimensions of three-layer residual network learning units. It has fewer network characteristics than two-layer residual network units by a factor of 17.35.

There are issues with the improved algorithms and the inception component. Although its enhanced techniques are so complex that the number of variables and computations can be overwhelming, overfitting frequently occurs, and the initial inception modules have little effect on improving network efficiency. The depth of the network is inadequate but its width is adequate. Low parameter effectiveness is caused by the mismatch among width and depth. There are some issues with the ResNet module as well. The amount of parameters and calculations rises quickly, even though the network is deepened and its classification accuracy is improved. Although the network structure is deeper but narrower, it is better than the inception component and upgraded methods.

Extraction of features with a worse diversity than the inception module results from an imbalance between breadth and depth. Due to the complexity of the residual component, a gradient explosion will result from the skip connection's raised training frequency being slower than the initial deceleration brought on by an abrupt rise in the number of variables and processes. The inception and the residual components can benefit from one another in several ways to boost recognition efficiency and reduce computations.

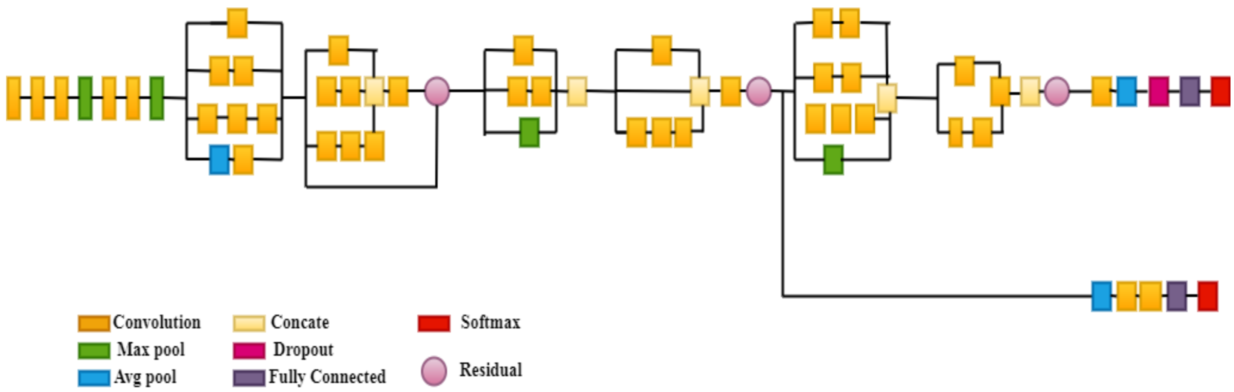


Figure 2. Framework of inception-ResNet-V2.

The core network of the model we propose is the inception-Resnet-V2 network. In Figure 2, the inception-Resnet-V2 architecture schematics are displayed.

3.3. Feature Selection

When irrelevant features are present in the data, the analysis method can be less accurate, and the model may learn those unwanted features. This issue is termed an optimization problem. The simplest method to avoid this problem is to use the analyzed dataset's optimal solutions. The Improved Reptile Search Algorithm (IRSA) is used to perform a feature selection procedure.

3.3.1. Reptile Search Algorithm

An innovative optimization technique called RSA imitates crocodiles' hunting and surrounding behavior. The crocodile is a semi-aquatic reptile with distinctive morphological traits, including a lined body that can raise one or both limbs to one face while walking. They can become skilled hunters in the wild owing to these traits.

The evolutionary algorithms RSA and ERSA were motivated by the way reptiles foraged for food. Their method of operation involves initializing a population of possible solutions, each of which represents a distinct subset of characteristics. Selection, crossover, and mutation are examples of operations that improve these solutions iteratively while simulating real evolutionary processes. The improved version, called ERSA, adds more mechanisms to improve the search process, which could speed up convergence and produce higher-quality solutions.

In the context of feature selection, the main optimization objective of RSA and ERSA is to determine the most pertinent subset of features that maximizes accurate classification while minimizing computational complexity. The algorithms seek to strike a compromise between minimizing the size of the feature set and preserving or enhancing the effectiveness of the classifier. This entails assessing possible solutions using a fitness function, which usually takes accuracy and the number of features chosen into account.

3.3.1.1. Initialization Phase

During this stage, chaotic maps applied to Equation (6) produce the initial potential solutions. The area of search and the objective function are also established. Additionally, each parameter's value is set during execution.

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,p} & x_{1,n-1} & x_{1,n} \\ x_{2,1} & \cdots & x_{2,p} & x_{2,n-1} & x_{2,n} \\ \cdots & \cdots & x_{o,p} & x_{i,n-1} & x_{i,n} \\ \vdots & \vdots & \vdots & \vdots & \\ x_{K-1,1} & \cdots & x_{K-1,j} & \cdots & x_{K-1,n} \\ x_{K,1} & \cdots & x_{K,j} & x_{K-1,n} & x_{K,n} \end{bmatrix} \tag{6}$$

While K denotes the amount of feasible results, n denotes the issues dimension, and the candidate's solution is denoted as X and $x_{0,p}$ denotes the ^{bth} agent of search location of the ^{ath} result.

$$x_{ij} = rand(UB - LB) + LB,\ \mathrm{P} = 1,2,\ldots,\mathrm{n} \tag{7}$$

While the value of initiation is indicated as *rand*. Additionally, *LB* and *UB* determine the lower and upper limits of the specified issue are provided.

3.3.1.2. Exploration Phase (Encircling)

Crocodiles use two techniques to encircle prey: belly walking and high walking. These motions pertain to several strategies dedicated to capturing the system's capacity for exploration. When they use a different hunting technique, the crocodile's motions hinder them from capturing the prey owing to their noise. Because of this, the exploratory search reveals a wide search zone; it can require several searches to find a promising site.

The RSA divided the entire amount of iterations into four phases and balanced the search between exploitation and exploration under four distinct circumstances. In RSA, there are two main approaches to search that are the focus of the exploration processes used to sift through the search space and identify superior solutions.

The spot upgrading equation is supplied for the exploration stage as stated in Equation (8).

$$x_{(a,b)}(t+1) = \begin{cases} Best_b(t) \times -\eta_{(a,b)}(t) \\ \times \beta - R_{(a,b)}(t) \times rand, \quad t \le \frac{T}{4} \\ Best_b(t) \times x_{r1,b} \times EH(t), \quad t \le 2\frac{T}{4} andt > \frac{T}{4} \end{cases} \tag{8}$$

While r and denotes a value among 0 and 1, t denotes the present iteration's amount, T denotes the maximum amount of iterations and $Best_b(t)$ displays the a^{th} spot for the best solutions. (a, b) indicates the operator in exploration. β inherited the input RSA.

$$\eta_{(a,b)} = Best_b(t) \times P_{(a,b)} \tag{9}$$

$$R_{(a,b)} = \frac{Best_b(t) - x_{(r2,b)}}{Best_b(t) + \varepsilon} \tag{10}$$

$$EH(t) = 2 \times r_3 \times \left(1 - \frac{1}{T}\right) \tag{11}$$

The value of $r2$ is a chance integer among [1, N] in the Equation (10), 2 is the coefficient of correlation employed to produce numbers among 2 and 0, and $r3$ denotes an integer.

$$P_{(a,b)} = \alpha + \frac{x_{(a,b)} - M(x_a)}{Best_b(t) \times (UB_{(b)} - LB_{(b)}) + \varepsilon} \tag{12}$$

The mean spots of the *ath* solution, as determined by Equation (12) are represented by $M(x_i)$. α is an important parameter that, in this work, has been set at a value of 0.1 and serves as a reference for the precision of the exploration undertaken as part of the cooperative hunting process.

$$M(x_i) = \frac{1}{n}\sum_{j=1}^{n} x_{(i,j)} \tag{13}$$

3.3.1.3. Phase in Exploitation (Hunting)

This stage introduces the exploitative behavior of the Algorithm. Crocodiles use two techniques during their hunting procedure: coordination and collaboration. These tactics mimic the exploitation search described in Equation (14). The original RSA shows the spot upgrading equation for the exploitation below.

$$x_{(a,b)}(t+1) = \begin{cases} Best_b(t) \times P_{(a,b)}(t) \times rand, \quad t \le 3\frac{T}{4} and\ t > 2\frac{T}{4} \\ Best_b(t) - \eta_{(a,b)}(t) \times \varepsilon \\ -R_{(a,b)}(t) \times rand, \quad \mathrm{t} \le \mathrm{Tandt} > 3\frac{T}{4} \end{cases} \tag{14}$$

$Best_b(t)$ denotes the best result identified thus far and (i, j) denotes the hunting parameter. $P(a, b)$ is the best-found solution's *jth* location.

3.3.1.4. Circle Chaotic Map

In order to maximize the diversity of initialized solutions, optimization algorithms frequently make use of chaos theory. The diversity of the population stands for potential solutions, components of a solution, or a framework that is easily convertible into a solution. According to the previous literature review, the optimization algorithm used in this case is a population-based algorithm, which begins problem solving by initializing a random solution and then begins evaluating it in light of the predetermined criteria. Using a search agent is necessary to initialize this solution; in this work, the search agent is the reptiles.

The search agents on the primary RSA algorithm begin with a position at random and produce random solutions. There is a population diversity problem because these haphazard solutions are seen as contributing to population diversity. The position of this search agent is set in this study using the circle Chaotic map function.

Algorithm performance is improved through the use of chaotic maps to enhance initialized solutions. Furthermore, compared to random search, chaos theory can delve deeper into the search space. But it's crucial to use solution space to its fullest in order to maximize the effectiveness of the initial population. In order to boost population diversity, this research initializes the IRSA using the Circle Map (CM) proposed by Chaos theory. A one-dimensional function taken from the circle themselves is the CM.

In mathematical terms, it is equivalent to identifying the angle of a point in the circle by calculating it's modulo 2π value, assuming that the point is in the circle line. When two numbers are divided by one another, a similar remainder is obtained as modulo of the two numbers. The result still shows an angle when modulo is computed with a value other than 2π, but it needs to be normalized so that the entire range between [0, 2π] is represented.

The CM control variables in this implementation are set to a=0:5 and b=0:2. The CM's mathematical model is calculated using the formula in the Equation (10).

$$ChaosCircleMap = x_{n+1} = x_{ij} + b - \left(\frac{a}{2\pi}\right) sin(2\pi x_{ij})\, mod(1), (0,1) \tag{15}$$

While *n* denotes the chaotic sequence *x*'s symbol and *xn* is the sequence's *nth* chaotic integer. As already stated, the regulating variables *b* and an aid in locating the chaotic effectiveness. The values from the Crocodiles' random beginning positions were substituted by the CM value in the IRSA.

3.3.1.5. Simulated Annealing

Several optimization techniques used the Simulated Annealing (SA) algorithm to enhance exploitative ability and guard against local search issues. Setting the parameters for the optimization methods. The SA is used in this study to enhance the optimal approach for the final of every iteration to overcome the optimal local stagnation issue of the original RSA. To avoid local optima, the worst option will be chosen with a well-defined likelihood and the best option is selected. The likelihood of selecting a less fortunate option, as shown in Equation (16), is determined by the Boltzmann likelihood function.

$$P = e - T(Gene_{Sol} - Best_{Sol}) \tag{16}$$

3.3.1.6. Enhanced Reptile Search Algorithm

With regard to the selection of features, this paper utilizes ERSA. Combining the raw RSA, the theory of chaos, and the strategy in SA, the IRSA has been presented. With this improvement, hope to improve categorization performance while using fewer characteristics overall. When employed to resolve high-dimensional issues like selecting features, the original RSA has two notable shortcomings. Local optima issues and various early remedies are some of these limitations. To solve the feature selection issue, two changes to the RSA are suggested. To increase the diversity of RSA solutions, the initial setup stage of the first upgrade integrates chaotic maps, particularly the CM.

When initializing the RSA population locations in the ERSA method, the CM value will replace the stochastic values. The Circle chaotic map serves as the source of the chaotic values. To boost the IRSA's exploitation capabilities, the second upgrade combines the SA with it. After each RSA repetition, SA enhances the best solution discovered after CM implementation.

3.3.1.7. Fitness Function

When an algorithm converges and accepts the current solution as the best one, it stops iterating. Typically, the criteria involve accomplishing a predetermined fitness threshold, going over a maximum number of iterations, or identifying a lack of discernible progress over multiple iterations. Calculating every solution's capacity for categorization and the number of chosen features in this study relies on the suggested fitness function. Nevertheless, the fitness value is computed once the candidate solution is initialized and saved as the top result. A fitness function is calculated after exploring and using the current optimal position in each cycle. According to the presumption, the current location (solution) has a higher fitness value than the previous one. A neighborhood search is then carried out once the enhanced solution has been used instead of the best option. Repeating this procedure results in meeting the stopping conditions. Employed in Equation (17) is the suggested fitness function.

$$Fitness = \alpha\gamma_R(D) + \beta\frac{R}{N} \tag{17}$$

While $\alpha\gamma_R(D)$ the utilized errors in categorization rate. *R* is the size of the chosen subset.

3.4. Sentiment Analysis Based on Novel Ensemble Approach

3.4.1. Deep Convolutional Neuro-Fuzzy Inference System

The Adaptive Neuro-Fuzzy Inference System (ANFIS) architecture is the foundation for our fuzzy analyzer. Since ANFIS is a universal approximator, it is conceptually equivalent to the fully connected layer(s) it replaces in the ResNet or LeNet structures. ANFIS has a layered design, and the first layer uses Equation (18) to calculate the membership of an input in a fuzzy set for fuzzy rule *i* and input *j*. For every rule, there is a single connection per input.

In traditional ANFIS, inputs are mapped via a feed-forward network to a layer of fuzzy rules, which generates outputs. This allows neural networks and fuzzy logic principles to be combined to model complex, nonlinear relationships. On the other hand, DCNFIS integrates deep convolutional layers, which are engineered to autonomously extract advanced features from the input data. This improves the system's capacity to manage intricate, multi-dimensional data that is commonly encountered in sentiment analysis assignments.

Because of its more straightforward structure, ANFIS may have trouble handling high-dimensional data, but DCNFIS's deep convolutional layers are built to handle this kind of data well, enabling it to handle and analyze massive amounts of sentiment data more skillfully. Furthermore, although ANFIS is capable of modeling nonlinear relationships, it might not be as good at expressing complex sentiments. The DL feature of DCNFIS is better equipped to handle the nuances and complexity of sentiment expressions found in social media and other text data sources because it can recognize and learn from complex patterns in the data. Furthermore, DCNFIS, with its DL foundation, is built to scale effectively with larger datasets, offering improved performance and accuracy in sentiment classification tasks, whereas ANFIS may experience scalability issues with large datasets.

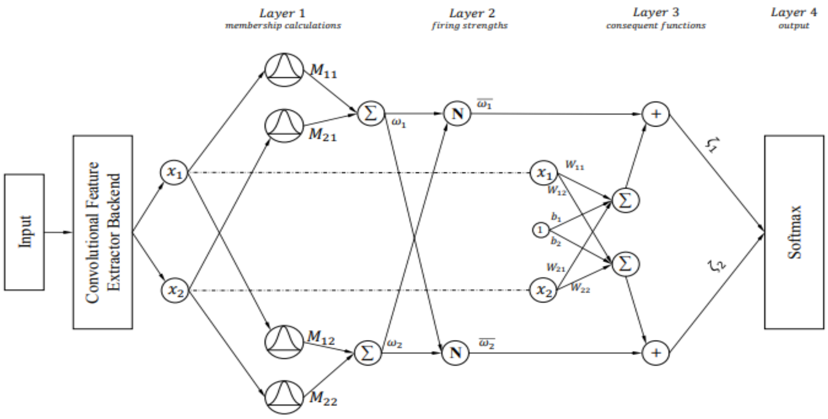


Figure 3. Block diagram of DCNFIS.

Figure 3 illustrates a system with two inputs, two fuzzy rules, and four functions for membership. To ensure mathematical stability in the tails, we estimate the logarithm of the natural number of the membership values. These calculations contrast our precise implementation with *ANFIS* calculations.

$$M_{ji} = exp(-\beta_{ji}.(x_j - \mu_{ji})^2) \quad \text{(ANFIS)}$$
$$M_{ji} = -0.5.\beta_{ji}.(x_j - \mu_{ji})^2 \quad \text{(DCNFIS)} \qquad (18)$$

The firing intensity of every fuzzy rule is determined by layer 2. Nevertheless, since we employ logarithms, we total the log memberships instead of the typical product of all preceding membership functions.

$$\omega_i = \Pi M_{ji} \quad \text{(ANFIS)}$$
$$\omega_i = \sum_i^j M_{ji} \quad \text{(DCNFIS)} \qquad (19)$$

In layer 3, the activation is typically calculated as the input variables' linear product divided by the corresponding rule's standardized firing intensity. In its place, a linear pairing of the input variables and the average of the log firing intensity is used.

$$\varsigma i = \bar{\omega}_i.\left(\sum W_{ji} \cdot x_j + b_i\right) \quad \text{(ANFIS)}$$
$$\varsigma i = \bar{\omega}_i + \left(\sum_j^i W_{ji} \cdot x_j + b_i\right) \quad \text{(DCNFIS)} \qquad (20)$$

With this update, we can produce fuzzy areas (clusters) and a fully trainable deep backend without concern about incorrect gradient explosions. Layer 4 then uses the softmax activation function to estimate the class likelihood.

$$P(c_i|x) = y_i = \frac{exp(\varsigma_i)}{\sum_0 exp(\varsigma_0)} \qquad (21)$$

A Fuzzy Inferential System (FIS), a rule-based expert system, operates similarly to how this architecture is designed to work. The layer transfer functions, in particular, simulate the steps in the processing that a FIS goes through to infer an outcome from its inputs. The fuzzy rule base can be inferred from a dataset using the ANFIS hybrid machine learning rule. A trained ANFIS can be directly converted into fuzzy rules because there is a 1:1 correlation between ANFIS and the FIS it mimics. This is simply true for ANFIS's outcomes from the last CNN layer, which are its direct inputs.

Each rule's network inputs are combined linearly in layer 3 of the ANFIS. In the fundamental ANFIS, the rule is weighed according to its firing power before being sent to a summation. Because we used the logarithm of the network signals in our execution, we added the log of the firing strength to the weighted sum in layer 3 before passing the sum to the layer 4 softmax function. The ranks of the softmax results are supposed to be identical to those of the initial ANFIS, even though the logit values were altered because the logarithm function is monotonically growing.

3.4.1.1. Learning Algorithm

Using the BER optimization algorithm, our fuzzy categorizer will learn the network's adaptive parameters. The parameters like an epoch, learning rate, Batch size, regularization parameters, Dropout rate, and weight are optimized by the BER algorithm.

3.4.1.2. Guided Backpropagation

The "deconvolution approach" has a new iteration for showing the characteristics that CNNs have learned: guided backpropagation. The negative gradient values are set to zero when guided backpropagation allows only the positive gradients to flow.

3.4.2. Multi-Layer Stacked Bidirectional LSTM Neural Network

During the training phase, the BiLSTM neural network resolves the accumulated error issue. Moreover, a bidirectional neural network built on the DL system is integrated to create the network. The proposed comprises a multi-layer forward framework and a multilayer reverse framework. The method depth is increased by the multi-layer stacked bidirectional LSTM neural network. To gain extensive knowledge of the data traits and enhance sentiment evaluation precision, input information can be frequently learned.

Each pair of layers in the multi-layer stacked architecture consists of a forward and a reverse LSTM neural network. The final results from the first forward and reverse LSTM layers are added and sent to the second layer. The forward and backward outcomes of every layer decide the network's outcome, and its representation can be described as follows.

$$o_t = g\left(V^{(j)} s_t^{(i)} + V^{(i)} s_t^{(i)}\right) \tag{22}$$

$$s_t^{(i)} = f\left(U^{*(i)} s_t'^{(-1)} + W^{(i)} s_{t+1}'\right) \tag{23}$$

$$s_t^{(1)} = f(U^{(1)} x_t + W^{(1)} s_{t-1}) \tag{24}$$

$$s'^{(1)}_t = f\left(U'^{(1)} x_t + W'^{(1)} s_{t-1}'\right) \tag{25}$$

In the ith concealed layer, whereas $s_t^{(i)}$ and $s_t'^{(-1)}$are the state of variables at t−1 and t time, accordingly. The weight data is not shared between the forward and reverse computations. The weight matrices among each input layer, concealed layer, and the final layer are denoted by the symbols (i), (i), and (i). The equivalent matrix inverse of weights in the reverse calculation is represented by the values of $V'(i)$, $U'(i)$, and $W'(i)$, respectively. The amount of bidirectional LSTM layers is denoted by i, while the output layer value is represented by i=0, 1, 2⋯∞.

3.5. Al-Biruni Earth Radius Optimization Algorithm

The BER optimization technique, presented in the present study, is motivated by the calculation of the earth radius for determining the region of search surrounding the outcomes in the cooperative behavior of swarm individuals to realize their worldly goals. With BER, an effort is made to balance promoting speedy convergence and preventing optimal local stagnation. That is achieved by applying the approach that enhances exploitation effectiveness, strikes an appropriate equilibrium among exploitation and exploration, expands the exploration of search space, and increases diversity among the current population.

3.5.1. Inspiration

Al-Biruni calculated the Earth's radius in the 11th century. The duration between the horizontal and the ground was determined from a hilltop. The equation that followed allowed him to determine the height of the hill:

$$h = \frac{d \tan\theta_1 \tan\theta_2}{\tan\theta_2 - \tan\theta_1} \tag{26}$$

Al-Biruni also climbed the mountain's crest to determine how far the horizon dipped. Utilizing his data and the following calculation to determine the Earth's radius,

$$R = \frac{h \cos\alpha}{1-\cos\alpha} \tag{27}$$

The Al-Biruni methodology mimics swarm cooperation and a global optimization objective. Swarms live in communities and cooperate to accomplish their objectives. They typically cooperate to gather food and protect themselves from invaders, trading off duties as necessary. They are divided into smaller groups, with people working together to accomplish their main objective within their smaller group and with members of other groups. An example of swarm collaboration is found in ant and bee colonies. Every swarm member contributes something unique to the colony. Workers go out and acquire food for the colony's fighters.

3.5.2. Exploration-Exploitation Balance

The number of people in every group is changed dynamically when the individual is separated into subcategories in the suggested model, improving the balance among the tasks of exploration and extraction. In order to explore and utilize the population, the procedure begins by splitting it into two categories. Compared to the exploitation group, which makes up 30% of the population, the exploration group makes up 70% of the total. While the initial limit is determined at 70%, the amount of people in the exploration group is reduced over iterations to 30%. Furthermore, if no superior solution is identified, the elitism technique is used to maintain the phase-leading solution, which ensures that the optimization process will converge for the population. The BER optimization technique may have an optimal local location if a solution's fitness is not substantially enhanced after three iterations.

3.5.3. Exploration Operation

As shown below, exploration avoids local optima standstill and advances toward the perfect solution while also identifying intriguing regions in the search space.

The paper presents the BER optimization algorithm and illustrates how the exploration group size gradually decreases during its iterative process. The convergence criteria play a critical role in optimization algorithms such as BER, as they dictate when the algorithm should stop or modify its parameters. Convergence criteria are typically based on a number of different things, like finishing a certain number of iterations, improving the objective function to a certain extent, or stabilizing the values of important variables over time.

3.5.4. Heading towards the Best Solution

The exploration team uses this method to identify probable places near the place it is now located in the search field. This is done by continually looking for the option with the highest fitness value among the available options that are close by. To do this, the BER research employs the formulas that follow:

$$r = h\frac{cos(x)}{1 - cos(x)} \tag{28}$$

$$\vec{D} = \vec{r}_1 \cdot (\vec{S}(t) - 1) \tag{29}$$

$$\vec{S}(t+1) = \vec{S}(t) + \vec{D} \cdot (2\vec{r}_2 - 1) \tag{30}$$

3.5.5. Exploitation Phase

It is responsible for enhancing previously established results. The BER computed the fitness values of every participant and then selected the best candidate. The subsections below detail the two methods the BER uses for attaining exploitation.

3.5.6. Heading towards the Best Solution

An exploration agent is directed toward the ideal solution using the subsequent computations:

$$\vec{S}(t+1) = r^2(\vec{S}(t) + \vec{D}) \tag{31}$$

$$\vec{D} = \vec{r}_3(\vec{L}(t) - \vec{S}(t)) \tag{32}$$

While r_3 is a vector of random generated by Equation (32) that directs the movement of stages in the direction of the best answer.

3.5.7. Examining the Region Near the Best Solution

The most appealing section is the one that includes the best response (the leader). As a result, some individuals look for alternatives near the best one in the hopes of finding one that is superior. The BER performs this process using the subsequent computation.

$$\vec{S}(t+1) = r(\vec{S}*(t) + \vec{k}) \tag{33}$$

$$\vec{k} = z + \frac{2 \times t^2}{N^2} \tag{34}$$

S stands for the ideal solution, *t* stands for the iteration amount, *N* stands for the sum of iterations, and *z* is an arbitrary integer among [0, 1]. An illustration of the mining and exploring processes.

3.5.8. Mutation Operation

The BER also conducts research using the mutation. One or more components of a person may be disrupted locally and probabilistically. A modification in the search field that helps avoid local optimization minimizes rapid convergence and serves as a jumping off point for another intriguing subject. The mutation operator is represented below.

$$\vec{S}(t+1) = \vec{k} * z^2 - h\frac{cos(x)}{1 - cos(x)} \tag{35}$$

3.5.9. Best Solution Selection

The Algorithm selects the optimal result to guarantee the quality of the identified results. The elitism approach improves computational performance but can prematurely converge multimodal functions. Due to its tremendous exploratory skills, the BER can prevent early convergence.

4. Result and Discussions

In this section, indicators of performance, including recall, accuracy, and precision, are briefly examined. Preprocessing, feature extraction, feature selection, and analysis are all a part of the broader investigation of TSA, which is also evaluated and fully described. Tables and bar graphs illustrate the findings by comparing the current and trending analyzers. The research was completed by importing the results of the evaluated metrics after briefly detailing the full technique.

The model may frequently yield lower accuracy or even overfitting or underfitting. It is essential to perform hyper-parameter tuning to achieve high model efficiency. Consequently, the hyper-parameter was adjusted and the accuracy was maximized using the randomized search technique. The hyper-parameter values in the suggested model are described in Table 2.

Table 2. Parameter setup.

Parameter	Value
Batch size	64
Epoch	200
Learning rate	0.0001
Dropout rate	0.3
Weight	0.9
threshold	0.5

4.1. Experimental Setup

An Intel i5 2.60 GHz processor and 16 GB of RAM operate Windows 10 on this device. The experiments are carried out in the Anaconda3 environment with Python, Keras, and Tensor Flow.

4.2. Dataset Description

Three datasets for sentiment analysis Sentiment140, SST5, and Twitter US Airline Sentiment are utilized in this study.

- **Twitter US Airline Sentiment Dataset**

In February 2015, CrowdFlower gathered and made the Twitter US Airline Sentiment dataset available. The dataset consists of 14640 tweets classified into neutral, negative, and positive. The dataset has 3099 neutral feedback, 9178 negative, and 2363 positive feedback. Six American airlines United, US Airways, South-West, Delta, and Virgin America are mentioned in the tweets in the dataset.

- **Sentiment140 Dataset**

Stanford University used Twitter to gather the Sentiment140 dataset. With 0.8 million positive and 0.8 million negative reviews, the dataset is evenly balanced.

The dataset has six columns, which are target, id, text, flag, user, and ID.

- **SST5 Dataset**

Treebank SST contains the labels of the emotions for 215,154 phrases in 11855 sentences. The set of data can be given as a binary categorizing, either positive or negative, or it may be displayed in a more precise manner using a 5-class categorize: negative, extremely negative, neutral, very positive, and positive.

4.3. Evaluation Metrics

Accuracy, F1-score, precision, and recall are a few of the evaluation measures employed in the suggested research.

- Accuracy: data values are successfully anticipated is indicated by the accuracy metric.

$$Accuracy = \frac{No.of correctpredictions}{Sum \text{ of prediction}} \tag{36}$$

- Recall: it determines the number of test case instances out of all the positive categories that are accurately predicted. The computation appears to be this:

$$Re\,c\,all = \frac{No.\text{ of True positives}}{\text{No. of true positives + No of False Negatives}} \tag{37}$$

- Precision: for all the anticipated positive category instances, the precision measure determines how many samples are truly positive:

$$Pr\,e\,cision = \frac{\text{No of True Positives}}{\text{No. of true } positives + \text{No of False Positives}} \tag{38}$$

- F1-score: precision and sensitivity are harmonically summed to obtain this value. The dice similarity coefficient and sorensen-dice coefficient are other names for it. 1. it's the ideal number. The following is an example of how the F1-score is calculated.

$$F1 - Score = 2 * \frac{Pr\,ecision * Re\,call}{Pr\,ecision + Re\,call} \tag{39}$$

4.3.1. GUI Implementation

Graphical User Interface (GUI) initial phase is shown in Figure 4. Here we enter the pharse and select the dataset.

Figures 5, 6, and 7 shows the instance for Word cloud, show tweets and analyze.

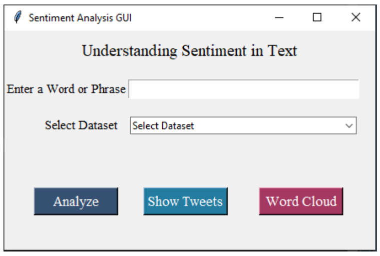


Figure 4. Initial phase of the GUI interface.

Figure 5. Example word cloud.

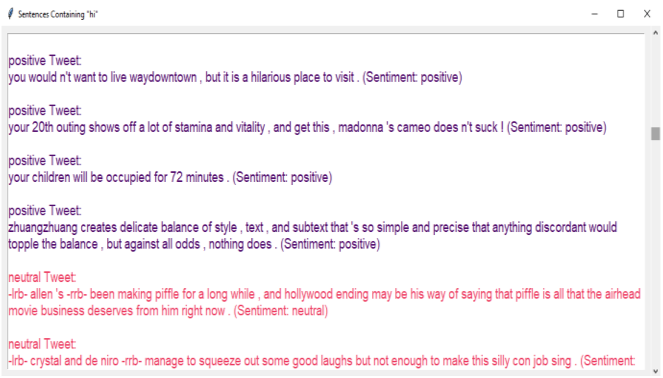


Figure 6. Instance for show tweets.

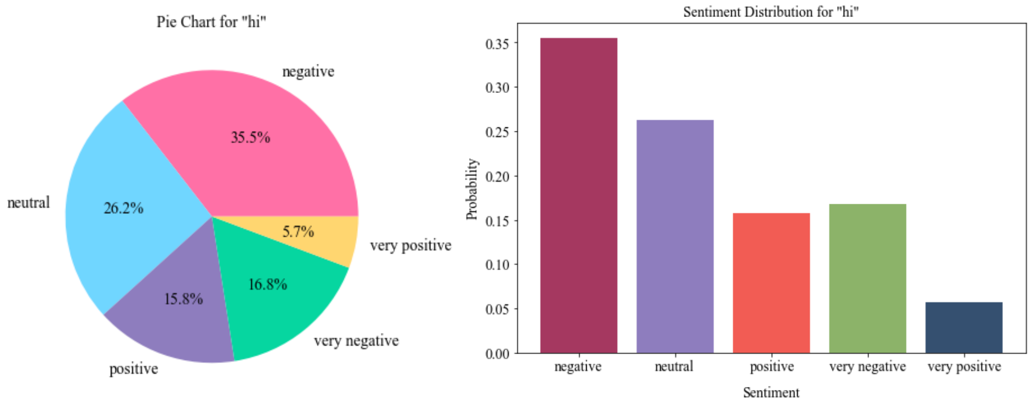


Figure 7. Analysis of tweets.

4.3.2. A Comprehensive Examination of Dataset 1 (Twitter US Airline Sentiment Dataset)

This portion utilizes the Twitter US Airline dataset to show how the proposed model performs compared to the latest approaches. For comparison with our suggested strategy, the following methods are used: LSTM, Gated Recurrent Unit (GRU), AdaBoost, CNN-BiLSTM, BiLSTM, and RoBERTa-LSTM.

Table 3. Comparison of existing DL approaches with proposed using dataset 1.

Approaches	F1-score (%)	Precision (%)	Recall (%)	Accuracy (%)
CNN-BiLSTM	67	70	65	77.32
LSTM	69	71	69	77.56
BiLSTM	70	71	69	77.46
GRU	72	73	71	78.55
AdaBoost	65	67	63	74.59
RoBERTa-LSTM	91	91	91	91.37
Proposed	98.77	99	98.56	99.02

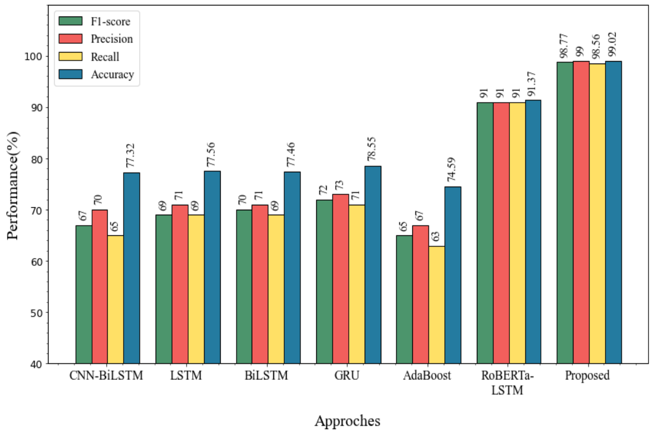


Figure 8. Analysis of DL approaches using dataset 1.

Existing DL approaches with proposed approach is compared in Table 3. By using dataset 1 our proposed approach gain superior performances. The second best performance was obtained in RoBERTa-LSTM approach. An graphical illustration of existing over proposed using datset 1 is shown in Figure 8.

Table 4. Comparison of existing Machine learning approaches with proposed using dataset 1.

Approaches	F1-score (%)	Precision (%)	Recall (%)	Accuracy (%)
KNN	60	60	60	68.41
NB	45	79	44	69.5
Decision tree	58	62	56	71.14
Logistic regression	72	78	69	80.5
Proposed	98.77	99	98.56	99.02

Comparison of existing machine learning approaches with proposed using dataset 1 is shown in Table 4 and Figure 9. The existing machine learning approaches like KNN, NB, decision tree and logistic regression are employed to compare with our proposed DL approaches. While contrasting with machine learning approaches, the proposed DL approach obtained superior performances of 98.77% F1-score, 99% of precision, 98.56% of recall, and 99.02% of accuracy.

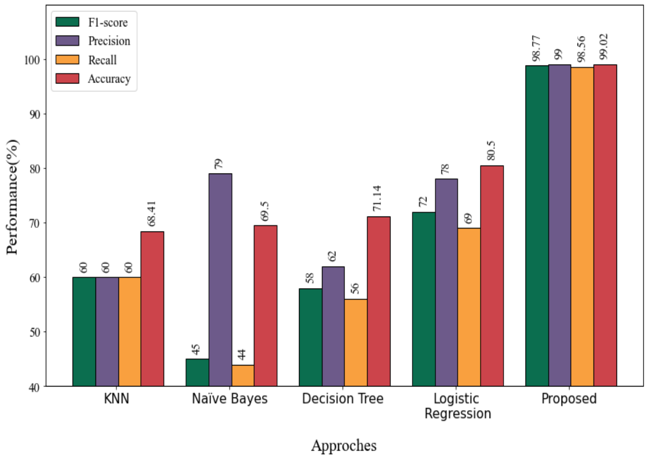


Figure 9. Comparison of machine learning over proposed deep fuzzy approach using dataset 1.

4.3.3. A Comprehensive Examination of Dataset 2 (Sentiment 140 Dataset)

Employing the Sentiment 140 dataset, this section compares the effectiveness of the suggested model against the latest approaches. Similarly, the already-in-use methods such as LSTM, GRU, AdaBoost, CNN-BiLSTM, BiLSTM, and RoBERTa-LSTM are used to compare with our suggested method.

Table 5 compares the proposed methodology to existing DL methods. Our suggested method performs better when dataset 2 is used. The RoBERTa-LSTM technique achieved the second-best efficiency. Figure 10 displays a graphical comparison of existing deep with the proposed using dataset 2.

Table 5. Comparison of proposed with existing DL approaches using dataset 2.

Approaches	F1-score (%)	Precision (%)	Recall (%)	Accuracy (%)
AdaBoost	69	71	70	69.94
CNN-BiLSTM	77	77	77	77.58
LSTM	79	79	79	79.10
GRU	78	78	78	78.96
BiLSTM	78	78	78	78.53
RoBERTa-LSTM	90	90	90	89.70
Proposed	98.88	98.90	98.87	99.03

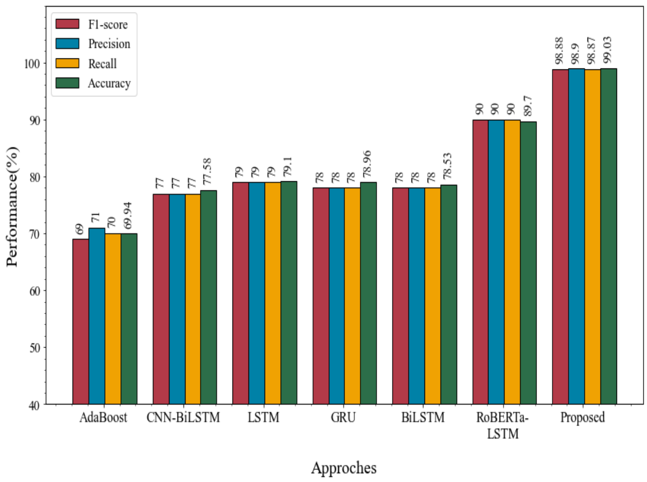


Figure 10. Comparison of existing DL with the proposed model using dataset 2.

Table 6. Comparison of existing machine learning with the proposed model using dataset 2.

Approaches	F1-score (%)	Precision (%)	Recall (%)	Accuracy (%)
Logistic regression	78	78	78	78.01
KNN	57	66	60	60.39
Decision tree	59	69	62	62.34
NB	77	77	77	76.57
Proposed	98.88	98.90	98.87	99.03

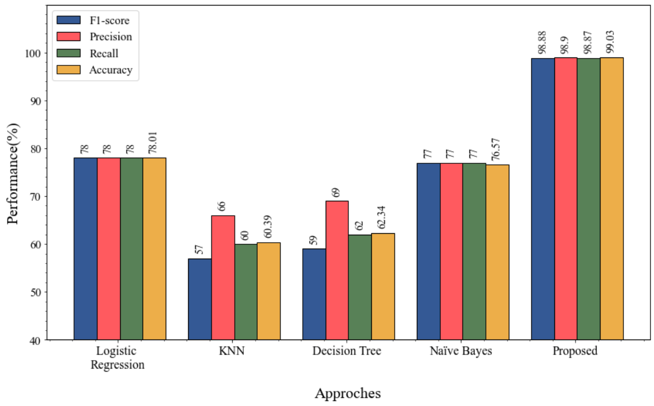


Figure 11. Differentiation of machine learning over proposed deep fuzzy approach.

Using dataset 2, Table 6 and Figure 11 compare proposed and existing machine learning techniques. We compare our suggested DL approaches with presently used machine learning techniques, such as KNN, NB, decision trees, and logistic regression. Compared to machine learning techniques, the suggested DL strategy achieves higher efficiency with a 98.88% F1-score, a 98.90 % precision, a 98.87% recall, and a 99.03% accuracy.

4.3.4. A Comprehensive Examination of Dataset 3 (SST5)

Employing the SST5 dataset, this section compares the effectiveness of the suggested model against the latest approaches. The existing approaches like DSA, SentiBERT, and BERT_LARGE are utilized to compare with the proposed approach.

Table 7. Comparison of existing approaches with proposed using dataset 3.

Models	F1-score	Precision	Recall	Accuracy
DSA	50.6	-	-	-
SentiBERT	30.7	-	-	-
BERT_LARGE	84.2	-	-	-
Proposed	99.01	99.06	98.97	99.11

Comparison of existing approaches with proposed using dataset 3 is shown in Table 7 and Figure 12. When compared to current methods, the suggested method produces a better result. Most of the existing approaches using dataset 3 cannot analyze the evaluation over F1-score, precision, recall and accuracy.

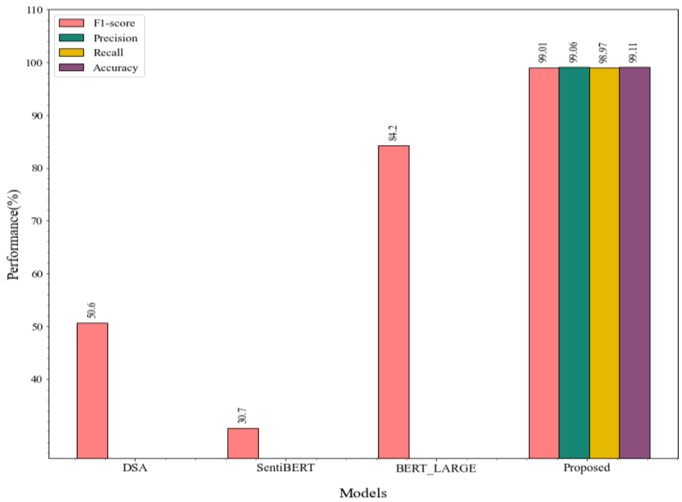


Figure 12. Comparison of existing approaches with proposed using dataset 3.

4.4. Discussion

The four processes used to carry out this suggested study are tweet preprocessing, feature extraction, feature selection, and sentiment analysis. Twitter data collecting is the first stage. The considered Twitter sentiment dataset is first put through preprocessing. Here, lemmatization, tokenization, stemming, bad word removal, and stop word removal. Following elimination, a clear preprocessed dataset is acquired. To achieve the desired target, the effectiveness of the suggested and current solutions is addressed in Table 7.

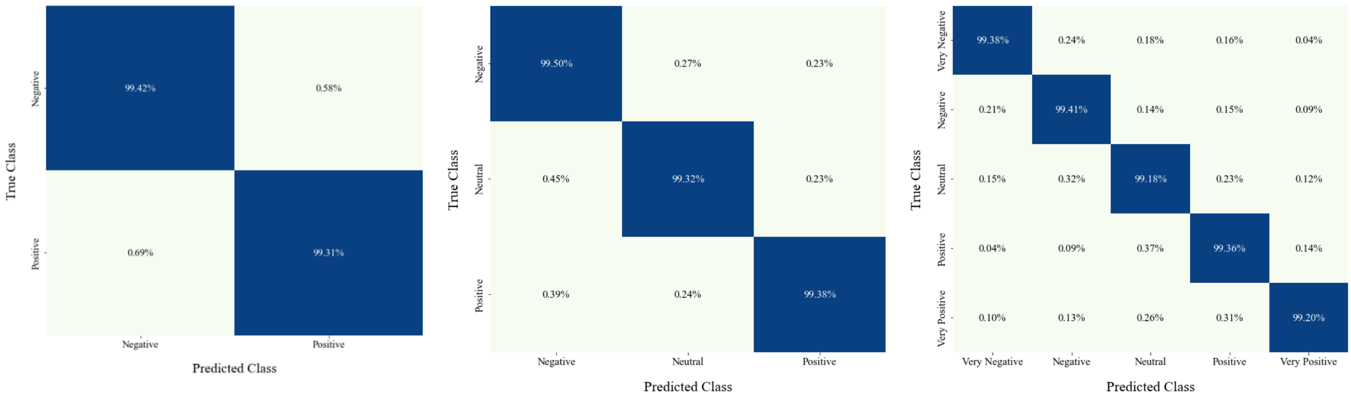
An inception-ResNet-V2 performs a feature extraction on this preprocessed dataset. The IRSA is the algorithm used to choose the features. The DCFNN approach is then employed to categorize the feelings expressed on Twitter. BER optimization algorithm is used to optimize the DCFNN Hyperparameters. Positive, negative, very positive, very negative, and neutral attitudes are all assessed. Analysis is done on the effectiveness of both the proposed classifier with a feature selection approach and the effectiveness of classifiers that are presently in use.

Using a ensemble of novel DL and fuzzy technique, we extensively executed our sentiment assessment method in our research. A wide range of online platforms' contributions from users served as the training dataset for the algorithm. We compared our approach to the most advanced sentiment analysis techniques like random forests and more conventional machine learning models like NB, in order to produce an accurate comparison. To maximize the effectiveness of our algorithm, we ran thorough experiments using multiple datasets, used standard metrics for evaluation, and modified hyperparameters. The contrast reveals how well our suggested approach captures sentiment nuances and how robust it is when compared to other techniques.

Table 8 depicts the effectiveness contrast among the suggested and current approaches. The information already available is compiled from earlier studies that used the Twitter dataset to construct sentiment analysis.

Table 8. Analyzing proposed performance with and with out optimization.

Datasets	Without optimization				With optimization			
	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Dataset 1	98.42	98.05	98.21	98.12	99.02	99	98.56	98.77
Dataset 2	98.10	98.16	98	98.07	99.08	98.90	98.87	98.88
Dataset 3	98.23	98.47	98.21	98.33	99.01	99.06	98.97	99.01



a) Sentiment 140. b) SST5. c) Twitter US Airlines.

Figure 13. Confusion matrix.

Table 9. Comparison over traditional approaches.

Approaches	Recall (%)	Precision (%)	F1-score (%)
LSTM	88.03	88.46	88.24
NB	84.13	86.02	85.06
CNN	84.12	85.10	84.61
RNN	86.60	87.18	86.89
BiLSTM	92.82	91.54	92.18
Proposed	98.97	99.06	99.01

A confusion matrix is a vital tool in the evaluation of the performance of a machine learning model, including sentiment analysis models. It's beneficial for tasks involving classification, such as sentiment analysis. The confusion matrix is a table that helps visualize and quantify the performance of a model by comparing its predictions to the actual labels in the dataset shown in Figure 13. Table 9 shows the differentiation over existing approaches.

The differentiation between existing traditional methods and the proposed approach is shown in Table 8. The existing approaches like LSTM, CNN, RNN, NB, and BiLSTM are employed to compare with the proposed approach. While differentiating from the proposed approach, the proposed approach yields a greater performance (98.97% recall, 99.06% precision and 99.01% F1-score).

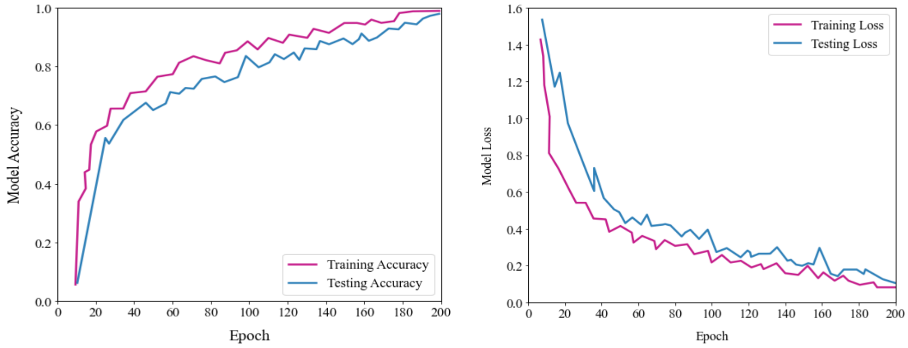
Several DL methods simply employ one feature extraction method, Term Frequency (TF) and Distinguishing Feature Selector (DFS), which does not effectively extract the features. The level of precision of the model that is suggested may be reduced by the techniques suggested without optimization. The fact that the feature extraction method in our proposed study can extract features from frequently used terms in the document contributes to its superior performance. The study suggested employing an optimization approach to raise the designed accuracy of the model level. Table 10 shows the differentiation of prior approaches with proposed.

Table 10. Comparison over prior approaches.

References	Methods	Dataset used	Accuracy	Precision	Recall	F1-score
Basiri *et al.* [21]	ABCDM	App, movie, kindle, sentiment 140	96.87	96.96	96.87	96.91
Jing *et al.* [22]	LSTM	Sentiment 140	79	79	79	79.10
Onan *et al.* [23]	CNN-LSTM	Twitter corpus	93.85	-	-	-
Behera *et al.* [24]	CNN-LSTM	Movie review, Airline review, US				
Liao *et al.* [25]	RoBERTa	Presidential election review and self-driving car review	89.70	78	78	78
Jain *et al.* [26]	DCNN-BERT	Twitter, movie review	97.89	95.13	94.67	95.06
Proposed			99.13	99.06	98.97	99.01

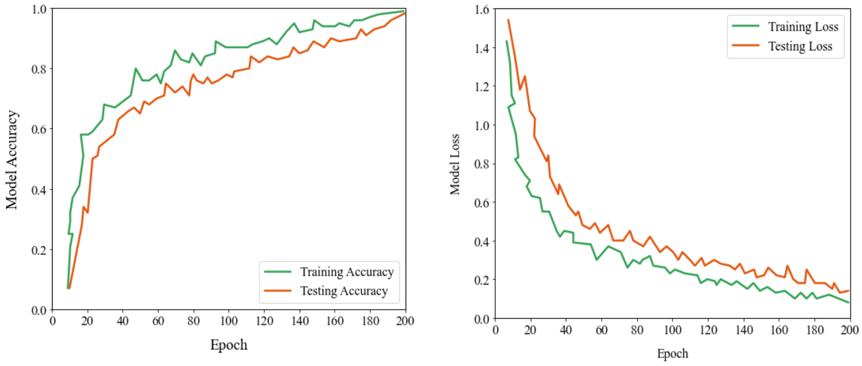
4.5. Evaluation of Training and Testing

A learning rate of 0.0001 is used to increase accuracy. By increasing the amount of training epochs, the algorithm's accuracy can be enhanced. We trained the framework additionally on 200 epochs. Figures 14, 15, and 16 illustrate the training and testing loss and accuracy.



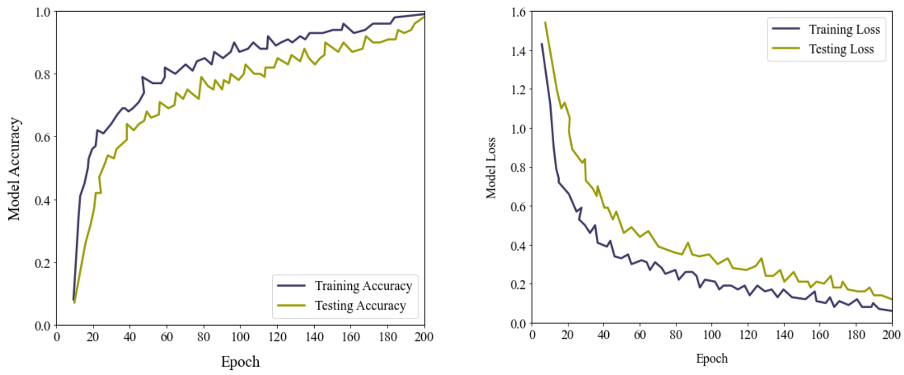
a) Training accuracy vs testing accuracy. b) Training loss vs testing loss.

Figure 14. Differentiation of dataset 1.



a) Training accuracy vs testing accuracy. b) Training loss vs testing loss.

Figure 15. Differentiation of dataset 2.



a) Training accuracy vs testing accuracy. b) Training loss vs testing loss.

Figure 16. Differentiation of dataset 3.

4.6. Ablation Study

The ablation study for the suggested model is discussed in Table 11. This describes the performance of the entire design, and Table 11 also provides a comparison of the various strategies now in use. Out of all the methods, the suggested strategy outperformed other algorithms in terms of performance efficiency. When all approaches are combined rather than used separately, the overall efficiency is increased, as shown by the analysis of the outcomes produced by these approaches and the integrated methods individually. Additionally, the feature selection method ablation study is assessed, and the findings are reported in Table 11.

Table 11. Result analysis related to the ablation experiment.

Methods	Models	Accuracy	Precision	Recall	F1-score
Classification	GARN	97.86	96.65	96.76	96.70
	GRN	96.10	95.67	95.77	94.86
	ARN	95.67	94.90	94.34	92.75
	RNN	93.19	94	93.9	91.52
	Proposed	99.13	99.06	98.97	99.01
Feature selection	HMWSO	97.86	95.99	96.67	95.89
	WSO	96.10	96.75	95.9	94.35
	MO	95	94.3	94.2	95.7
	Proposed	99.03	98.89	98.76	98.47

The suggested model's definition of complexity in computation is as follows: The level of complexity of the attention framework is On^2 • d, the complexity of the recurrent network is On• d^2, and the complexity of the gated recurrent network is Ok • n • d^2. The suggested approach is Ok • n^2 • d in terms of overall complexity. This complexity demonstrates that the suggested model's effective performance was achieved by lowering system complexity. The model will not function well enough when used alone, though. But using such models in combination has produced better results than using other techniques now in use.

4.7. Limitations and Future Scope

Although the suggested strategy has delivered adequate results, it hasn't yet attained the required level. This is because the proposed architecture did not give the selected elements equal weight. As an outcome, the proposed design is less successful because certain important features are missed. In order to identify visual emotion utilizing all the chosen features, we will introduce an effective DL approach in the future together with the best selection of features approach. Additionally, the effectiveness of the present system is being examined using a tiny dataset; however, in the future, large data sets with more challenging images will be employed.

5. Conclusions

Sentiment analysis is essential for understanding public opinion in a variety of fields, including business and politics, so that strategic decisions may be taken. As a result, an efficient algorithm is needed for identifying the polarity (positive, negative, very positive, very neutral or neutral) of the opinions dynamically. To investigate different perspectives of Twitter users who use the online platform, optimized ensemble approach is utilized in the present investigation. We ensemble novel DCNFIS and DL based multilayer stacked bidirectional LSTM neural network to analyzes the sentiments. The hyperparameters are optimized by BER optimization algorithm. The optimized novel approach enhance the performance. The relevant features are selected and extracted with the help of and inception-ResNet-V2 method. In sentiment analysis, feature selection and feature extraction are useful strategies because they enhance model effectiveness, efficiency, and interpretability. Twitter US Airlines, Sentiment 140, and SST5 datasets were used in the development process. Recall, precision, F-measure, and accuracy, are the four indicators of performance used to evaluate the effectiveness of the proposed approach with those of other DL and machine learning models, including GRU, Bi-LSTM, KNN, logistic regression, LSTM, NB, and RoBERTa-LSTM. The experimental investigations stated that the proposed approach gain superior performance over existing approaches.

References

[1] Basiri M., Nemati S., Abdar M., Asadi S., and Rajendra Acharrya U., "A Novel Fusion-based Deep Learning Model for Sentiment Analysis of COVID-19 Tweets," *Knowledge-based Systems*, vol. 228, pp. 107242, 2021. https://doi.org/10.1016/j.knosys.2021.107242

[2] Basiri M., Nemati S., Abdar M., Cambria E., and Rajendra Acharya U., "ABCDM: An Attention-based Bidirectional CNN-RNN Deep Model for Sentiment Analysis," *Future Generation Computer Systems*, vol. 115, no. C, pp. 279-294, 2021. https://doi.org/10.1016/j.future.2020.08.005

[3] Behera R., Jena M., Rath S., and Misra S., "Co-LSTM: Convolutional LSTM Model for Sentiment Analysis in Social Big Data," *Information Processing and Management*, vol. 58, no. 1, pp. 102435, 2021. https://doi.org/10.1016/j.ipm.2020.102435

[4] Chauhan P., Sharma N., and Sikka G., "The Emergence of Social Media Data and Sentiment Analysis in Election Prediction," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 2601-2627, 2021. https://link.springer.com/article/10.1007/s12652-020-02423-y

[5] Dashtipour K., Gogate M., Adeel A., Larijani H., and Hussain A., "Sentiment Analysis of Persian Movie Reviews Using Deep Learning," *Entropy*, vol. 23, no. 5, pp. 1-16, 2021. https://doi.org/10.3390/e23050596

[6] Dashtipour K., Gogate M., Cambria E., and Hussain A., "A Novel Context-Aware Multimodal Framework for Persian Sentiment Analysis," *Neurocomputing*, vol. 457, pp. 377-388, 2021. https://doi.org/10.1016/j.neucom.2021.02.020

[7] Elgeldawi E., Sayed A., Galal A., and Zaki A., "Hyperparameter Tuning for Machine Learning Algorithms Used for Arabic Sentiment Analysis," *Informatics*, vol. 8, no. 4, pp. 1-21, 2021. https://doi.org/10.3390/informatics8040079

[8] Fatouros G., Soldatos J., Kouroumali K., Makridis G., and Kyriazis D., "Transforming Sentiment Analysis in the Financial Domain with ChatGPT," *Machine Learning with Applications*, vol. 14, no. 1, pp. 100508, 2023. https://doi.org/10.1016/j.mlwa.2023.100508

[9] Garcia K. and Berton L., "Topic Detection and Sentiment Analysis in Twitter Content Related to COVID-19 from Brazil and the USA," *Applied Soft Computing*, vol. 101, pp. 107057, 2021. https://doi.org/10.1016/j.asoc.2020.107057

[10] Iddrisu A., Mensah S., Boafo F., Yeluripati G., and Kudjo P.,"A Sentiment Analysis Framework to Classify Instances of Sarcastic Sentiments within the Aviation Sector," *International Journal of Information Management Data Insights*, vol. 3,

no. 2, pp. 100180, 2023. https://doi.org/10.1016/j.jjimei.2023.100180

[11] Jain D., Boyapati P., Venkatesh J., and Prakash M., "An Intelligent Cognitive-Inspired Computing with Big Data Analytics Framework for Sentiment Analysis and Classification," *Information Processing and Management*, vol. 59, no. 1, pp. 102758, 2022. https://doi.org/10.1016/j.ipm.2021.102758

[12] Jain P., Quamer W., Saravanan V., and Pamula R., "Employing BERT-DCNN with Sentic Knowledge Base for Social Media Sentiment Analysis," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 8, pp. 10417-10429, 2023. https://doi.org/10.1007/s12652-022-03698-z

[13] Jing N., Wu Z., and Wang H., "A Hybrid Model Integrating Deep Learning with Investor Sentiment Analysis for Stock Price Prediction," *Expert Systems with Applications*, vol. 178, pp. 115019, 2021. https://doi.org/10.1016/j.eswa.2021.115019

[14] Kaur G. and Sharma A., "A Deep Learning-based Model Using Hybrid Feature Extraction Approach for Consumer Sentiment Analysis," *Journal of Big Data*, vol. 10, no. 1, pp. 1-23, 2023. https://doi.org/10.1186/s40537-022-00680-6

[15] Kaur H., Ul Ahsaan S., Alankar B., and Chang V., "A Proposed Sentiment Analysis Deep Learning Algorithm for Analyzing COVID-19 Tweets," *Information Systems Frontiers*, vol. 23, no. 6, pp. 1417-1429, 2021. https://doi.org/10.1007/s10796-021-10135-7

[16] Li R., Chen H., Feng F., Ma Z., Wang X., and Hovy E., "Dual Graph Convolutional Networks for Aspect-based Sentiment Analysis," *in Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, Bangkok, pp. 6319-6329, 2021. https://aclanthology.org/2021.acl-long.494.pdf

[17] Li W., Shao W., Ji S., and Cambria E., "BiERU: Bidirectional Emotional Recurrent Unit for Conversational Sentiment Analysis," *Neurocomputing*, vol. 467, no. C, pp. 73-82, 2022. https://doi.org/10.1016/j.neucom.2021.09.057

[18] Liang B., Su H., Gui L., Cambria E., and Xu R., "Aspect-based Sentiment Analysis via Affective Knowledge Enhanced Graph Convolutional Networks," *Knowledge-based Systems*, vol. 235, pp. 107643, 2022. https://doi.org/10.1016/j.knosys.2021.107643

[19] Liao W., Zeng B., Yin X., and Wei P., "An Improved Aspect-Category Sentiment Analysis Model for Text Sentiment Analysis Based on RoBERTa," *Applied Intelligence*, vol. 51, no. 6, pp. 3522-3533, 2021. https://doi.org/10.1007/s10489-020-01964-1

[20] Maree M., Eleyat M., and Mesqali E., "Optimizing Machine Learning-based Sentiment Analysis Accuracy in Bilingual Sentences via Preprocessing Techniques," *The International Arab Journal of Information Technology*, vol. 21, no. 2, pp. 257-270, 2024. https://doi.org/10.34028/iajit/21/2/8

[21] Mujahid M., Lee E., Rustam F., Washington P., Ullah S., Reshi A., and Ashraf I., "Sentiment Analysis and Topic Modeling on Tweets about Online Education during COVID-19," *Applied Sciences*, vol. 11, no. 18, pp. 1-25, 2021. https://doi.org/10.3390/app11188438

[22] Neogi A., Garg K., Mishra R., and Dwivedi Y., "Sentiment Analysis and Classification of Indian Farmers' Protest Using Twitter Data," *International Journal of Information Management Data Insights*, vol. 1, no. 2, pp. 100019, 2021. https://doi.org/10.1016/j.jjimei.2021.100019

[23] Nezhad Z. and Deihimi M., "Twitter Sentiment Analysis from Iran about COVID 19 Vaccine," *Diabetes and Metabolic Syndrome: Clinical Research and Reviews*, vol. 16, no. 1, pp. 102367, 2022. DOI:10.1016/j.dsx.2021.102367

[24] Onan A., "Sentiment Analysis on Product Reviews Based on Weighted Word Embeddings and Deep Neural Networks," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 23, pp. 1-12, 2021. https://doi.org/10.1002/cpe.5909

[25] Singh M., Jakhar A., and Pandey S., "Sentiment Analysis on the Impact of Coronavirus in Social Life Using the BERT Model," *Social Network Analysis and Mining*, vol. 11, no. 1, pp. 1-33, 2021. https://link.springer.com/article/10.1007/s13278-021-00737-z#Sec14

[26] Sitaula C., Basnet A., Mainali A., and Shahi T., "Deep Learning-based Methods for Sentiment Analysis on Nepali COVID-19-Related Tweets," *Computational Intelligence and Neuroscience*, vol. 2021, no. 1, pp. 1-11, 2021. https://doi.org/10.1155/2021/2158184

[27] Talaat A., "Sentiment Analysis Classification System Using Hybrid BERT Models," *Journal of Big Data*, vol. 10, no. 1, pp. 1-18, 2023. https://doi.org/10.1186/s40537-023-00781-w

Potu Bharath received his M-Tech Degree from ANU, Guntur. He is a full time Ph.D. Research Scholar with the School of Computer Science and Engineering, VIT-AP University, Amaravati. He has 4 years of teaching experience. Her research interests include Deep Learning, Machine Learning, and Natural Language Processing.

Dasari Venkata Lakshmi received master's and Ph.D. degrees from the University of Hyderabad, Hyderabad, India, in 1993 and 2008, respectively. Her Ph.D. research project titled ''On Vector Valued Amalgam Spaces.'' She was with the Faculty of Mathematics, Bapatla Engineering College, Bapatla, India, 1996-2021. Currently, she has been a Professor with the School of Computer Science and Engineering, VIT-AP University, India, since 2021. She has more than 24 years of teaching experience and 12 years of research experience. She has published more than 35 research articles in national and international journals. Her research interests include Machine Learning, Artificial Intelligence, Graph Theory, and Theoretical Computer Science.