

Article

Low-Resource Language Processing Using Improved Deep Learning with Hunter–Prey Optimization Algorithm

Fahd N. Al-Wesabi ^{1,*} , Hala J. Alshahrani ², Azza Elneil Osman ³ and Elmouez Samir Abd Elhameed ⁴

¹ Department of Computer Science, College of Science & Art at Mahayil, King Khalid University, Abha 62529, Saudi Arabia

² Department of Applied Linguistics, College of Languages, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia

³ Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam Bin Abdulaziz University, Al-Kharj 16278, Saudi Arabia

⁴ Department of Computer Science, College of Post-Graduated Studies, Sudan University of Science and Technology, Khartoum 11111, Sudan

* Correspondence: falwesabi@kku.edu.sa

Abstract: Low-resource language (LRL) processing refers to the development of natural language processing (NLP) techniques and tools for languages with limited linguistic resources and data. These languages often lack well-annotated datasets and pre-training methods, making traditional approaches less effective. Sentiment analysis (SA), which involves identifying the emotional tone or sentiment expressed in text, poses unique challenges for LRLs due to the scarcity of labelled sentiment data and linguistic intricacies. NLP tasks like SA, powered by machine learning (ML) techniques, can generalize effectively when trained on suitable datasets. Recent advancements in computational power and parallelized graphical processing units have significantly increased the popularity of deep learning (DL) approaches built on artificial neural network (ANN) architectures. With this in mind, this manuscript describes the design of an LRL Processing technique that makes use of Improved Deep Learning with Hunter–Prey Optimization (LRLP-IDLHPO). The LRLP-IDLHPO technique enables the detection and classification of different kinds of sentiments present in LRL data. To accomplish this, the presented LRLP-IDLHPO technique initially pre-processes these data to improve their usability. Subsequently, the LRLP-IDLHPO approach applies the SentiBERT approach for word embedding purposes. For the sentiment classification process, the Element-Wise–Attention GRU network (EWAG-GRU) algorithm is used, which is an enhanced version of the recurrent neural network. The EWAG-GRU model is capable of processing temporal features and includes an attention strategy. Finally, the performance of the EWAG-GRU model can be boosted by adding the HPO algorithm for use in the hyperparameter tuning process. A widespread simulation analysis was performed to validate the superior results derived from using the LRLP-IDLHPO approach. The extensive results indicate the significant superiority of the performance of the LRLP-IDLHPO technique compared to the state-of-the-art approaches described in the literature.

Keywords: low-resource languages; natural language processing; deep learning; sentiment analysis; hunter–prey optimizer

MSC: 68T50



Citation: Al-Wesabi, F.N.; Alshahrani, H.J.; Osman, A.E.; Abd Elhameed, E.S. Low-Resource Language Processing Using Improved Deep Learning with Hunter–Prey Optimization Algorithm. *Mathematics* **2023**, *11*, 4493. <https://doi.org/10.3390/math11214493>

Academic Editors: Victor Manuel Darriba Bilbao, Alexander Gelbukh and Alvaro Rodrigo

Received: 30 August 2023

Revised: 22 October 2023

Accepted: 27 October 2023

Published: 30 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Social media is one of the quickest ways for individuals to express themselves, leading to a flood of content on newsfeeds that reflects their opinions [1]. Analysing these newsfeeds is a direct method for capturing their sentiments and emotions. Sentiment analysis (SA), also known as opinion mining, is the process of identifying, extracting, and

categorizing specific information from unstructured texts using text analysis and computational linguistic techniques in Natural Language Processing (NLP) [2]. SA involves classifying opinionated textual content into polarity categories such as positive, negative, or neutral [3–6]. LRL processing has a profound effect on SA by extending the scope of languages that can be analysed. It facilitates the inclusion of languages with limited digital resources into SA application, thereby making sentiment analysis culturally diverse and more inclusive. This enables organizations to gain insights into sentiment trends, consumer preferences, and brand perception in previously underserved regions and languages, improving their global market understanding. Furthermore, LRL processing allows for cross-cultural analysis, supports humanitarian efforts in crisis responses, and contributes to the preservation of endangered languages, representing its wide-ranging implications for SA in our increasingly interconnected world.

LRLs, frequently spoken by underserved or marginalized populations, present unique challenges in the field of NLP. These languages lack the abundance of digital resources that are readily available for high-resource languages like English (such as large, labelled datasets and pre-trained models). Despite this scarcity in terms of resources, the significance of addressing LRL processing is vital for many compelling reasons. The major concern this research study aims to address is the limited accessibility and utilization of NLP technology for LRLs. This faces countless barriers, including the absence of well-established language technologies, the scarcity of labelled datasets, and limited linguistic resources. Subsequently, there is an urgent need for innovative approaches that make LRL processing more accessible, impactful, and effective.

The classification of text based on various features remains an interesting topic of study [7]. SA and opinion mining employ rule-based systems, deep learning (DL), and machine learning (ML) to continually enhance this area of research. Consequently, the advent of highly complex language models able to use previous knowledge and adapt it to the particular tasks in which it is utilized has improved performance and decreased the expenditure of computing resources [8]. Of particular interest are language models dependent upon deep neural networks (DNNs), which have the significant capability of classifying sentiments by automatically learning important features from databases [9]. However, these outcomes are highly dependent upon the language considered and especially on the accessibility of the extensive databases used to train the model in its early stages. This condition generally only applies to Chinese and English languages, while other languages are typically classified as LRLs [10].

This paper introduces LRL Processing using Improved Deep Learning with the Hunter–Prey Optimization (LRLP-IDLHPO). The LRLP-IDLHPO technique begins with data pre-processing to improve the usability of the data. Next, it applies the SentiBERT algorithm for word embedding purposes. Then, the sentiment classification process is performed by the Element-Wise–Attention GRU network (EWAG-GRU), an enhanced variant of the RNN. The HPO approach is applied for fine-tuning to further improve the performance of the EWAG-GRU algorithm. A comprehensive set of simulations was performed to validate and ensure the better performance of the LRLP-IDLHPO method. The key contributions of this paper are as follows:

- The LRLP-IDLHPO method, which confronts the challenges of LRL processing for SA by integrating SentiBERT, EWAG-GRU-based classification, and HPO-based parameter tuning is proposed. To the best of our knowledge, the proposed model has never been described previously in the literature.
- SentiBERT helps convert text data into a numerical representation that captures semantic context, enabling accurate SA in LRL settings.
- The EWAG-GRU model, an advanced variant of the RNN which can effectually process temporal features was employed, and its results improved using the attention mechanism for sentiment classification.

- The addition of the HPO technique for fine-tuning the EWAG-GRU algorithm illustrates a commitment to optimizing model performance, ensuring it operates at its best in the SA task.

2. Related Works

In [11], the authors proposed the extraction of sentiments from tweets dependent on their topical subject. The model employs NLP techniques for recognizing sentiments related to a specific problem. In this research study, three different methods were utilized to identify sentiments: classification depending on subjectivity, semantic association, and classification depending on polarity. AlBadani et al. [12] employed deep learning (DL) methods in various real-time applications across multiple domains, including sentiment analysis (SA). This study introduced an innovative and efficient approach to SA, utilizing DL techniques by integrating “universal language model fine-tuning” (ULMFiT) with a Support Vector Machine (SVM) to enhance recognition accuracy and effectiveness. Additionally, a novel DL method was employed for Twitter SA to recognize the opinions of individuals. Anand et al. [13] aimed to address MOLD_DL (Multilingual Offensive Language Detection using DL) approaches and utilized NLP in FS and classification. This FS was implemented to segment information using a fuzzy-based FCNN. Later, the extraction of chosen features and classification was executed by combining the model of the Bi-LSTM method with a hybrid NB framework with a SVM.

Kumar et al. [14] presented a technique that employed Graph Neural Networks (GNNs) for classifying texts based on their content. GNNs were implemented because they work effectively with 2D vectors, and through using GNNs, textual data can be represented in a 2D format. The computation of Self-Organizing Maps (SOM) was carried out to compute the adjacent neighbours in the graphs and determine the actual distances among the neighbours. In [15], the tweets of individuals were analysed using hybrid deep learning (DL) algorithms. SA was conducted through a five-point scale classification, which includes categories such as positive, negative, highly negative, highly positive, and neutral. This approach was found to require less time when handling a larger number of tweets compared to other methods, namely Decision Trees (DT), Random Forest (RF), and Naive Bayes (NB) classifiers. Alyoubi and Sharma [16] presented a new hybrid embedding technique aimed at augmenting word embeddings through the integration of NLP techniques. This study also introduced a novel DL algorithm for feature extraction and BiRNN for temporal and contextual feature application.

Rodrigues et al. [17] suggested a technique that can identify whether tweets are “ham” or “spam” and estimate the sentiment of tweets. The extracted features after pre-processing the tweets can be classified using different classifiers, such as LR, DT, multinomial NB, Bernoulli NB, RF, and SVM to detect spam, and these methods have been utilized for SA. Zuheros et al. [18] recommended the SA-based Multiperson Multicriteria Decision Making (SA-MpMcDM) technique for aiding smarter decisions. This involved combining an end-to-end multitask DL algorithm for feature-based SA, called the DOC-ABSADeepL approach, which was capable of detecting the feature classifications stated in an expert analysis and extracting their conditions and opinions.

3. The Proposed Model

In this manuscript, we propose the use of the LRLP-IDLHPO system for the processing of LRLs. The LRLP-IDLHPO technique enables the detection and classification of the different kinds of sentiments present in LRL data. To accomplish this, the presented LRLP-IDLHPO technique incorporates pre-processing, SentiBERT, a EWAG-GRU model, and a HPO algorithm for hyperparameter tuning. Figure 1 shows the overall flow of the LRLP-IDLHPO approach.

3.1. Data Pre-Processing

Data pre-processing phases differ based on the SA task and the features of the database. Executing suitable pre-processed approaches is vital to generating a clean and informative database that allows for correct sentiment forecasting via ML approaches. Text cleaning, tokenization, and lowercasing are employed to eliminate irrelevant noise from the text. However, techniques such as stemming and lemmatization further reduce words to their base forms, thereby enhancing the model’s ability to recognize sentiment-related patterns. Special attention is paid to the handling of emojis, negations, and emoticons that change sentiment context.

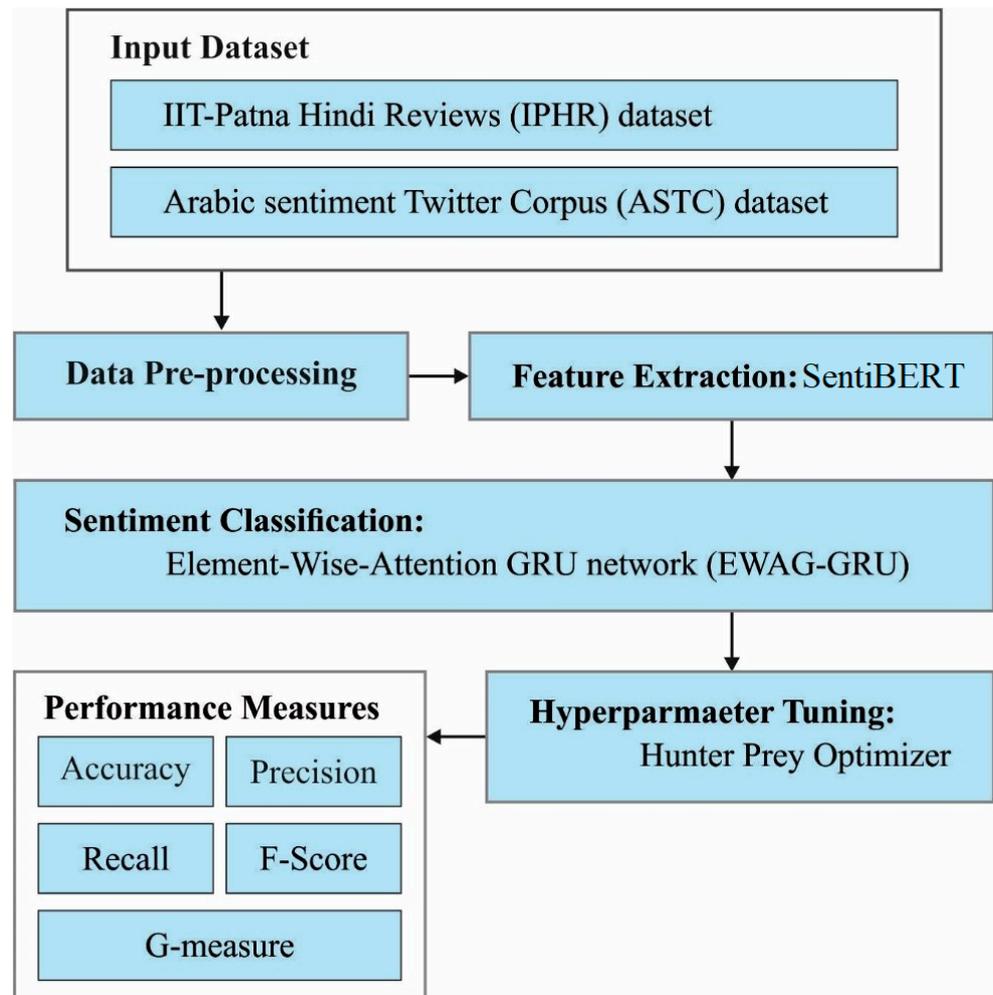


Figure 1. The overall flow of the LRLP-IDLHPO algorithm.

3.2. SentBERT Model

BERT is an attention-based language method that employs a stack of transformers encoded and decoded for learning textual data [19]. It also employs a multi-head attention mechanism for extracting helpful features for tasks. The bi-directional transformer NN, as the encoded feature of BERT, changes the entire word token to a numeric vector to process a word embedded for words that are semantically connected, which will be decoded to the numerically close embeddings. BERT and its variations are executed for several NLP tasks, including named entity detection, relation extraction, machine translation, and question and answer, accomplishing the desired outcomes.

The proposed approach employs the Sent-iBERT approach for word embedding. SentiBERT is instrumental in converting words or tokens from the LRL data into numerical representations that capture semantic and contextual information. This embedding process

is fundamental for the sentiment analysis task, as it enables the model to understand the meaning and sentiment associated with each word or phrase. SentiBERT adjusts BERT by adding a phrase node forecast unit and semantic composition unit. Specifically, the semantic composition unit’s purpose is to attain phrase representation, which is led by contextual word embedding and an attentive constituency parsing tree. SentiBERT tokenizes input text, generates contextualized word embeddings, and utilizes a classification head to predict sentiment labels. Its ability to capture context and adapt to LRLs makes SentiBERT a powerful tool for accurate sentiment analysis in languages with limited linguistic resources.

3.3. Design of the EWAG-GRU Model for Classification

The EWAG-GRU model is an enhanced version of the RNN used for the classification process, and it has the ability to process temporal features with the inclusion of an attention strategy [20]. Integrating the attention and gating mechanism in the DL model, we use an element-wise attention Gate (EQAG) to provide attention to the RNN neuron, which allows the RNN neuron to gain the capability to concentrate on the building blocks of input. It can apply a shareable EWAG with a similar size to the outcome attention vector as input to execute each neuron of the RNN blocks.

The RNN architecture better demonstrates the features of EWAG-GRU. The outcome response r_t of the t time step is evaluated in the input x_t , and the output r_t is as follows:

$$r_t = \tanh(W_{xr}x_t + W_{rr}r_{t-1} + \varepsilon_r) \tag{1}$$

In Equation (1), $a \in \{x, r\}$, $b \in \{r\}$, and $c \in \{r\}$, where W_{ab} represents the weight matrix for a and b , and ε_c represents the bias vector.

EWAG provides the aforementioned RNN neuron attention capability; a_t represents the response vector, and the dimension is similar to the prior RNN’s input x_t . The computation formula is as follows:

$$a_t = \varphi(W_{xa}x_t + W_{ra}r_{t-1} + \varepsilon_a) \tag{2}$$

In Equation (2), The significance level of the input \tilde{x}_t can be defined by the existing input r_{t-1} and the prior hidden layer (HL) x_t . φ represents the Sigmoid activation function. The input x is updated using the attentional response model as follows:

$$\tilde{x}_t = x_t \odot a_t \tag{3}$$

Then, the GRU model implements a recursive computation dependent upon upgraded input x . Figure 2 depicts the infrastructure of the GRU.

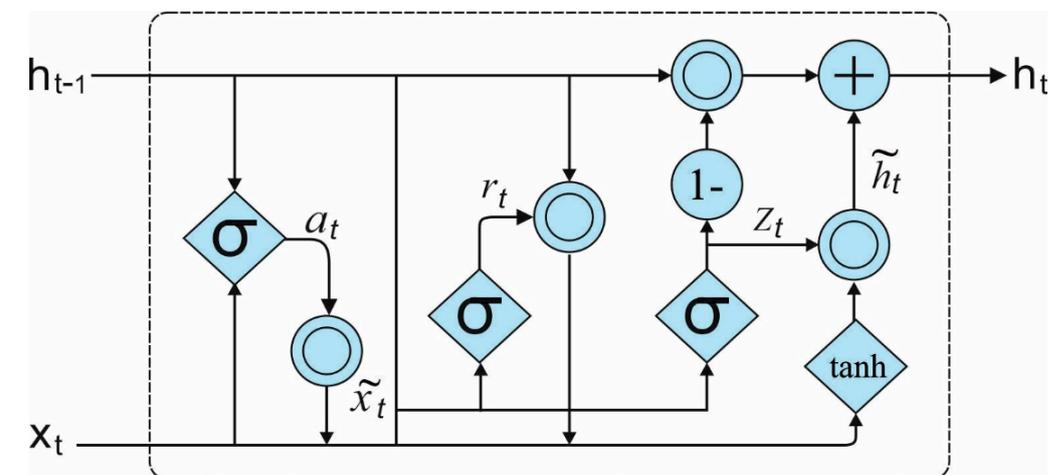


Figure 2. GRU structure.

The GRU is a kind of RNN that is developed to address the problem of long-term memory and vanishing gradients. It involves updating and resetting the gating units. The former defines what amount of the prior data is to be given to the existing state, whereas the latter controls the amount of novel input that needs to be integrated into the existing state. Once the EWAG was applied to the GRU block, it provided RNN neurons with the capability to selectively attend to the crucial components from the input series. The computation formula for the EWAG-GRU block is given below:

$$r_t = \sigma(W_r \odot [h_{t-1}, a_t \odot x_t] + \varepsilon_r) \tag{4}$$

$$z_t = \sigma(W_z \odot [h_{t-1}, a_t \odot x_t] + \varepsilon_z) \tag{5}$$

$$\tilde{h}_t = \tanh(W \odot [r_t \times h_{t-1}, a_t \odot x_t] + \varepsilon_h) \tag{6}$$

$$h_t = h_{t-1} + z_t (\tilde{h}_t - h_{t-1}) \tag{7}$$

where z_t represents the update gate, and r_t represents the reset gate. h_t denotes the output vector of HL. a_t refers to the response vectors, W represents the respective weight matrix, \tanh , and σ represents the activation function. ε denotes the bias vector, and \tilde{h}_t represents the vector after activation. Then, use the response of a_t , the EWAG, to control X_t to \tilde{X}_t and replace X_t with \tilde{X}_t to perform the follow-up. This is known as EWAG-GRU.

The network selectively focuses on the features significant to all the inputs, which analyse various components with different levels of attention to attain more specific outcomes. The network resolves the problems of reducing long-term dependency, along with the problems of time-series exclusion (produced via data analysis for managing the correlation). This contributed to an enhancement in detection performance amidst continuous activity.

3.4. Processes Involved in HPO-Based Hyperparameter Tuning

In this work, the HPO algorithm was utilized for the tuning of the hyperparameters related to the EWAG-GRU approach. The HPO algorithm is a new swarm-based optimizer technique that stimulates the behaviours among the prey and predators [21]. The HPO updates its features as it imitates the predictor behaviours but hunts the target; meanwhile, the target moves towards a safer position to escape from the predators. Consequently, the safer position is updated dynamically, and the predator needs to adapt its position according to the safer position. In HPO, Since the HPO is the metaheuristic algorithm, it begins with the group of random solutions that is calculated by the subsequent equation.

$$Z_j = lb + rand \times (ub - lb) \quad i = 1, 2, \dots, N, \tag{8}$$

In Equation (8), $rand \in [0, 1]$ represents the uniformly distributed random number. ub and lb denote the upper and lower boundaries of the searching region (vector form with dimensional = $1, 2, \dots, D$), and the N and D symbols are the overall size of populations and the amount of problem variables. The fitness function (FF) can be evaluated by the first set of solutions to identify the bad and good performances. Next, according to the fundamental steps of the HPA method, the initial phase of the solution is updated in the set of independent runs. During the exploration phase, the searching agent with a higher chance is used to determine the global and local points in the searching region. At the same time, the exploitation stage retakes the randomized minimum to circulate the potential solution. Iraj et al. developed the following equation for modelling the exploitation and exploration stages.

$$Z_{im}(t + 1) = Z_{im}(t) + 0.5 \left[\left(2\alpha\beta Prey_{P(m)} - Z_{im}(t) \right) + \left(2(1 - \alpha)\beta\mu_{(m)} - Z_{im}(t) \right) \right] \tag{9}$$

In Equation (9), $Z_{im}(t)$ and $Z_{im}(t + 1)$ denote the existing and future locations of the j^{th} hunter, respectively. The prey location is represented as $Prey_p$; the β , α , and μ

symbols are the balancing parameters, adaptive parameters, and mean of each location, respectively. These parameters can be calculated using the following equations:

$$R \text{ and } = \vec{R}_1 < \alpha; \text{index} = (Rand == 0);$$

$$\beta = R_2 \otimes \text{index} + \vec{R}_3 \otimes (\sim \text{index}). \tag{10}$$

$$\alpha = 1 - it \left(\frac{0.98}{MAXIt} \right). \tag{11}$$

where $Rand$, R_2 , \vec{R}_1 , and \vec{R}_3 indicate the random vector within $[0, 1]$, and $the \text{index}$ represents the index number of vector \vec{R}_1 that meets the conditions of $(P == 0)$. The balance variables α are calculated by Equation (11). The α operator has a value that declines from 1 to 0.02 in the iteration. $MAXIt$ refers to the maximum amount of iterations.

As previously stated, the aim is to catch the target; thus, the prey updates the position, employing the average of location (μ) using Equation (10), and later calculates the distance of all the searching agents from the mean location.

$$\mu = \frac{1}{n} \sum_{i=1}^n \vec{Z}_i. \tag{12}$$

The distance can be measured according to Euclidean distance

$$D_{euc(i)} = \left(\sum_{m=1}^d (Z_{im} - \mu_m)^2 \right)^{0.5}. \tag{13}$$

The searching agent with the maximum distance in the mean of placement is considered prey ($Prey_{P(m)}$) based on the following expression:

$$\overrightarrow{Prey_{P(m)}} = \vec{Z}_i \left| \text{issorted} D_{euc} (kbest) \right). \tag{14}$$

where $est = round(\alpha \times N)$ and N represents the solution counts. Once the target is attacked, it attempts to run away to escape towards the safer region. Iraj et al. considered the better safer position as the optimum global location, and the hunter updates its location to choose another target as follows:

$$Z_{im}(t + 1) = G_{P(j)} + \alpha \beta \cos(2\pi R_4) * (G_{P(j)} - Z_{im}(t)), \tag{15}$$

In Equation (15), G_P represents the optimum global position (safer location), and $R_4 \in [-1, 1]$ represents the random integer.

Flag =

$$\left\{ \begin{array}{l} Z_{im}(t + 1) = Z_{im}(t) + O.5 \left[\left(2\alpha \beta Prey_{P(m)} - Z_{im}(t) \right) + \left(2(1 - \alpha) \beta \mu_{(m)} - Z_{im}(t) \right) \right] \\ \text{if } R_5 \leq \gamma, \\ Z_{im}(t + 1) = G_{P(j)} + \alpha \beta \cos(2\pi R_4) * (G_{P(j)} - Z_{im}(t)) \\ \text{if else} \end{array} \right. \tag{16}$$

In Equation (16), R_5 represents the random integer within $[0, 1]$, and γ denotes the regulatory parameter with a value of 0.1.

The HPO algorithm derives an FF to attain enhanced classifier outcomes. It explains a positive integer to represent the best outcomes for the candidate performances. In this case, the minimized classification error rate is assumed as FF, as expressed in Equation (17).

$$fitness(x_i) = ClassifierErrorRate(x_i)$$

$$= \frac{\text{No. of misclassified instances}}{\text{Total no. of instances}} * 100 \tag{17}$$

4. Results and Discussion

The proposed model was simulated using the Python 3.10.10 tool with the following packages: tensorflow-gpu == 2.10.0, pandas, nltk, tqdm, scikit-learn, pyqt5, matplotlib, seaborn, gensim, prettytable, and numpy. The proposed model was experimented on using PC i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, 250GB SSD, and 1TB HDD.

The experimental validation of the LRLP-IDLHPO technique was tested on the IIT-Patna Hindi reviews (IPHR) [22] database and the Arabic Sentiment Twitter Classification (ASTC) [23] Database. This dataset was built to provide an Arabic sentiment corpus for the research community to investigate DL approaches for Arabic SA. The dataset includes tweets annotated with positive and negative labels. The dataset is balanced and consists of data that use positive and negative emojis. For experimental validation, we used 70% of the training dataset and 30% of the testing dataset.

The measures used to examine the performance of the proposed model were accuracy, precision, recall, F-score, and Geometric mean ($G_{measure}$) [24]. Figure 3 demonstrates the classifier analysis of the LRLP-IDLHPO system on the IPHR database. Figure 3a,b represent the confusion matrix achieved via the LRLP-IDLHPO technique at 70:30 of the TR set/TS set. The outcome value signified that the LRLP-IDLHPO method classified and detected all three classes accurately. Also, Figure 3c shows the PR curve of the LRLP-IDLHPO system. The outcome value specified that the LRLP-IDLHPO algorithm attained higher PR outcomes on three class labels. Figure 3d demonstrates the ROC study of the LRLP-IDLHPO methodology. The outcome showed that the LRLP-IDLHPO method resulted in effective experimental results, with higher ROC values on three classes.

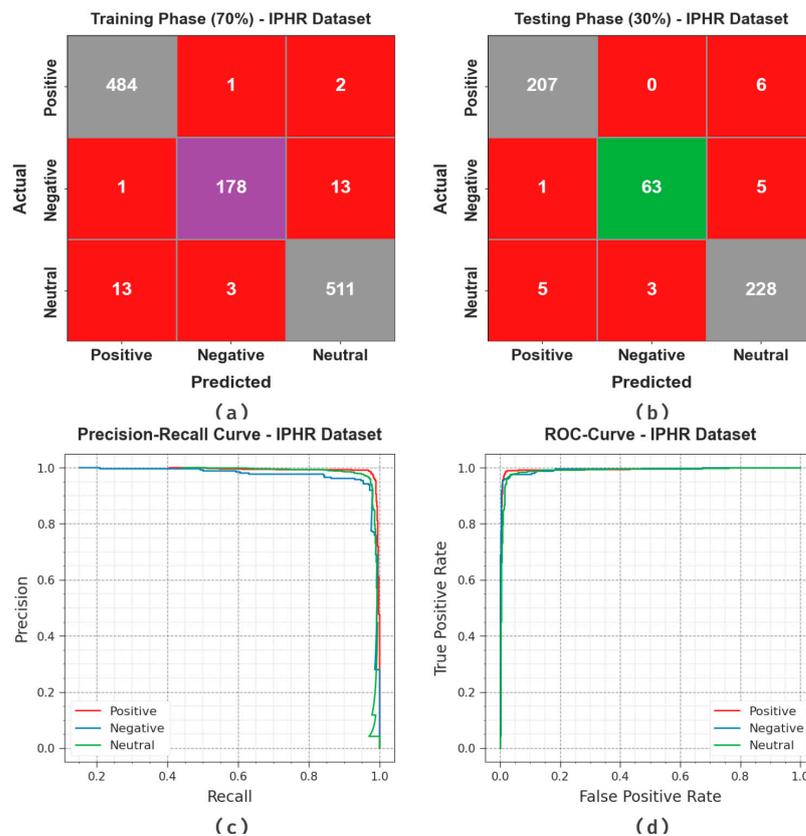


Figure 3. IPHR database: (a,b) confusion matrices, (c) PR_curve, and (d) ROC.

In Table 1 and Figure 4, the outcomes resulting from using the LRLP-IDLHPO technique on the IPHR database are provided. The table values imply that the LRLP-IDLHPO

technique properly recognizes three classes. Under the 70% TR set, the LRLP-IDLHPO technique reaches an effectual $accu_y$ of 98.18%, a $prec_n$ of 97.38%, a $reca_l$ of 96.35%, an F_{score} of 96.84%, and a $G_{measure}$ of 96.85%. Likewise, under the 30% TS set, the LRLP-IDLHPO technique attains an efficient $accu_y$ of 97.43%, a $prec_n$ of 96.01%, a $reca_l$ of 95.03%, an F_{score} of 95.51%, and a $G_{measure}$ of 95.51%.

Table 1. Classifier outcomes resulting from using the LRLP-IDLHPO algorithm on the IPHR Database.

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	$G_{Measure}$
TR set (70%)					
Positive	98.59	97.19	99.38	98.27	98.28
Negative	98.51	97.80	92.71	95.19	95.22
Neutral	97.43	97.15	96.96	97.06	97.06
Average	98.18	97.38	96.35	96.84	96.85
TS set (30%)					
Positive	97.68	97.18	97.18	97.18	97.18
Negative	98.26	95.45	91.30	93.33	93.36
Neutral	96.33	95.40	96.61	96.00	96.00
Average	97.43	96.01	95.03	95.51	95.51

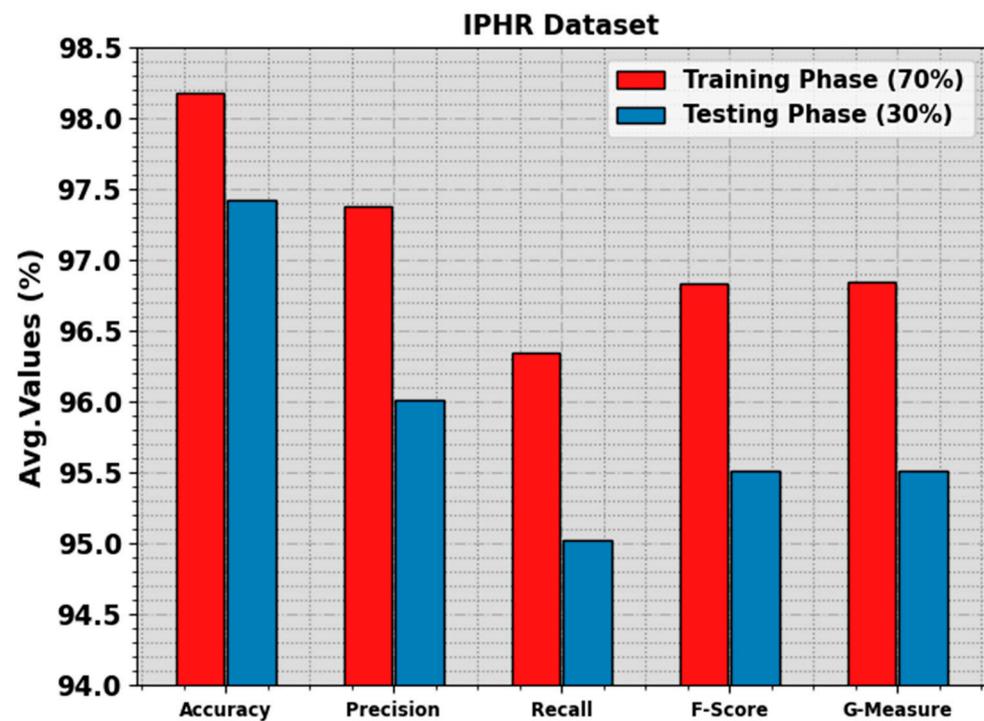


Figure 4. Average values derived from using the LRLP-IDLHPO algorithm on the IPHR Database.

Figure 5 shows the training accuracy (TR_{accu_y}) and validation accuracy (VL_{accu_y}) values derived from using the LRLP-IDLHPO system on the IPHR database. The TR_{accu_y} is defined by the estimation of the LRLP-IDLHPO method on the TR database, whereas the VL_{accu_y} is calculated by evaluating the performance of an individual testing database. The outcomes revealed that TR_{accu_y} and VL_{accu_y} rise with an increase in epochs. Therefore, the performance of the LRLP-IDLHPO algorithm improves on the TR and TS database with an increase in the number of epochs.

In Figure 6, the TR_{loss} and VR_{loss} curves derived from using the LRLP-IDLHPO system on the IPHR database are shown. The TR_{loss} determines the error between the predicted performance and original values on the TR data. The VR_{loss} measures the

performance of the LRLP-IDLHPO algorithm on separate validation data. The outcomes specified that the TR_loss and VR_loss tend to reduce with increasing epochs. It implies the improved performance of the LRLP-IDLHPO method and its ability to generate accurate classification. The decreased values of TR_loss and VR_loss indicate the superiority of the LRLP-IDLHPO system in capturing relationships and patterns.

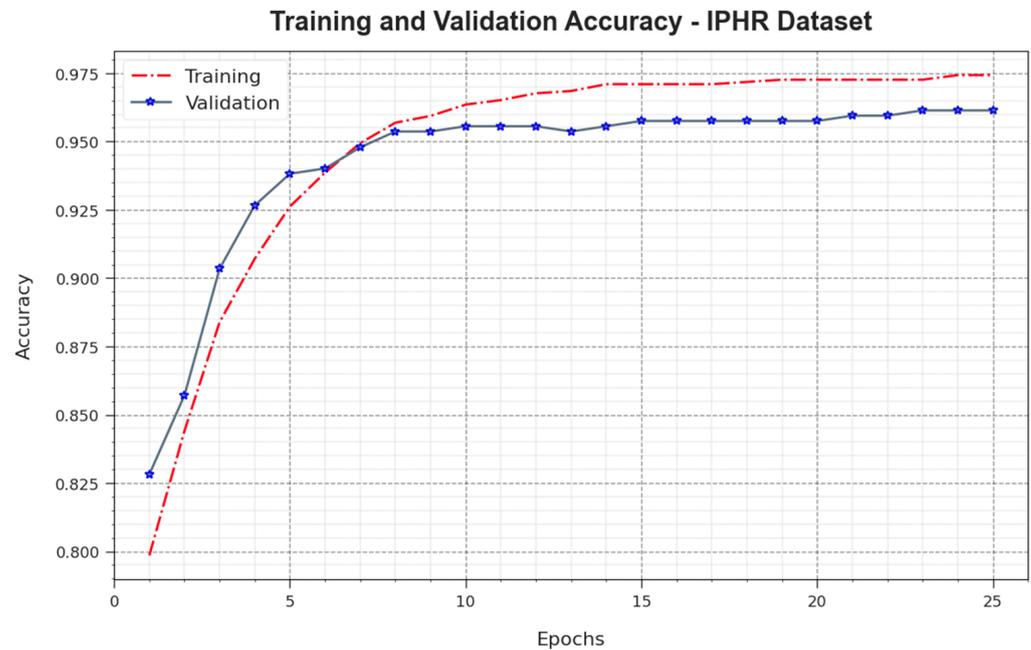


Figure 5. Accuracy curve derived from using the LRLP-IDLHPO algorithm on the IPHR Database.

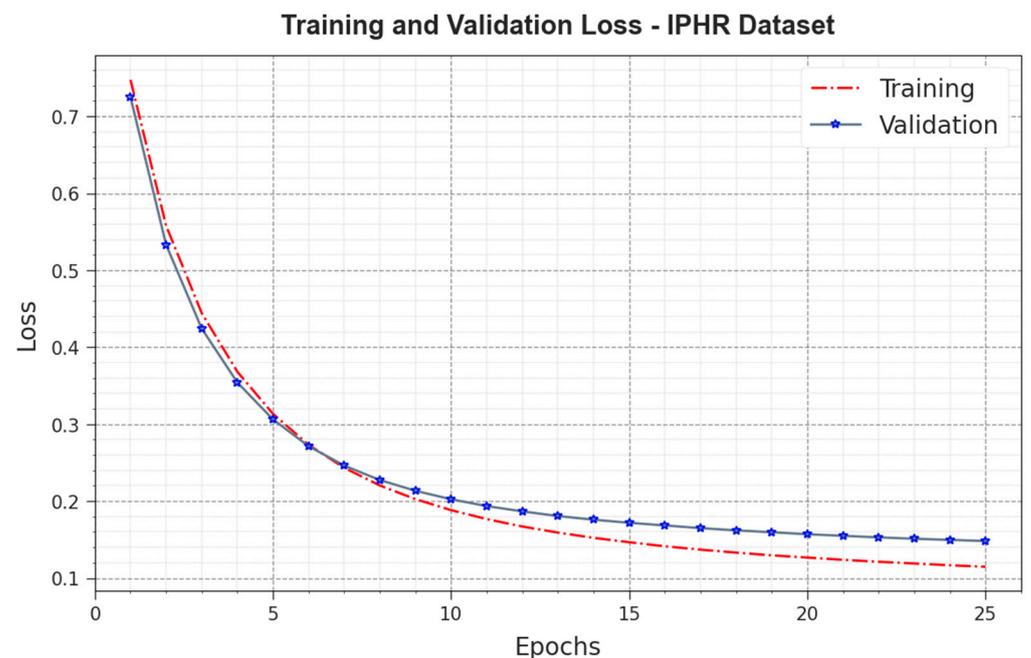


Figure 6. Loss curves derived from using the LRLP-IDLHPO algorithm on the IPHR Database.

A comparison of the results derived from using the LRLP-IDLHPO technique on the IPHR databases are reported in Table 2 and Figure 7 [24–26]. The results indicate that the NB approach yields worse outcomes, but the DT and LR approaches achieve closer values. Additionally, the RNN and GRU models yield reasonable performance. Although the LSTM and IAOADL-ABSA model achieve considerable results, the LRLP-IDLHPO

technique exhibits superior results, with maximum $accu_y$, $prec_n$, $reca_l$, and F_{score} values of 98.18%, 97.38%, 96.35%, and 96.84%, respectively.

Table 2. Comparison of the results derived from using the LRLP-IDLHPO system and other models on the IPHR Database.

IPHR Database				
Approach Models	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}
Decision Tree [24]	90.87	90.89	90.90	90.88
Logistic Regression [24]	92.94	93.05	93.02	92.95
Naive Bayes [24]	86.32	86.84	86.31	86.26
RNN [25]	95.09	95.08	95.09	95.10
LSTM Model [25]	97.12	95.16	95.06	95.17
GRU Model [25]	95.04	95.03	95.03	95.04
IAOADL-ABSA [26]	97.96	97.01	95.65	96.10
LRLP-IDLHPO	98.18	97.38	96.35	96.84

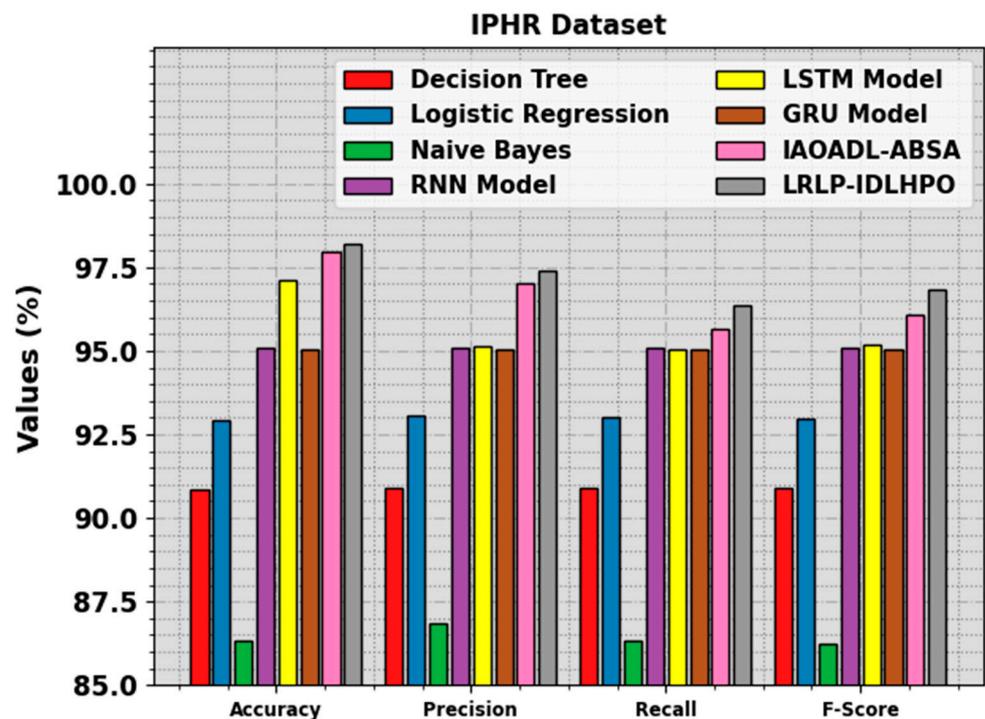


Figure 7. Graphical comparison of the results derived from using the LRLP-IDLHPO algorithm and other models on the IPHR Database.

Figure 8 illustrates the classifier performance of the LRLP-IDLHPO system on the ASTC database. Figure 8a,b depict the confusion matrix achieved by the LRLP-IDLHPO algorithm at 70:30 of the TR set/TS set. The results suggest that the LRLP-IDLHPO approach detected and classified all three classes accurately. Figure 8c depicts the results derived from the PR examination of the LRLP-IDLHPO approach. The simulation values suggest that the LRLP-IDLHPO approach achieved greater values of PR in three classes. However, Figure 8d demonstrates the ROC curve of the LRLP-IDLHPO approach. This result shows that the use of the LRLP-IDLHPO approach led to proficient performance in terms of ROC in three classes.

In Table 3 and Figure 9, the experimental outcomes derived from using the LRLP-IDLHPO algorithm on the ASTC database are provided. The values in this table imply that the LRLP-IDLHPO method properly recognizes three class labels. Under the 70% TR set, the LRLP-IDLHPO system achieves an effectual $accu_y$ of 99%, a $prec_n$ of 99%, a $reca_l$ of 99%,

an F_{score} of 99%, and a $G_{measure}$ of 99%. Similarly, under the 30% TS set, the LRLP-IDLHPO approach achieves an efficient $accu_y$ of 99%, a $prec_n$ of 99%, a $reca_l$ of 99%, an F_{score} of 99%, and a $G_{measure}$ of 99%.

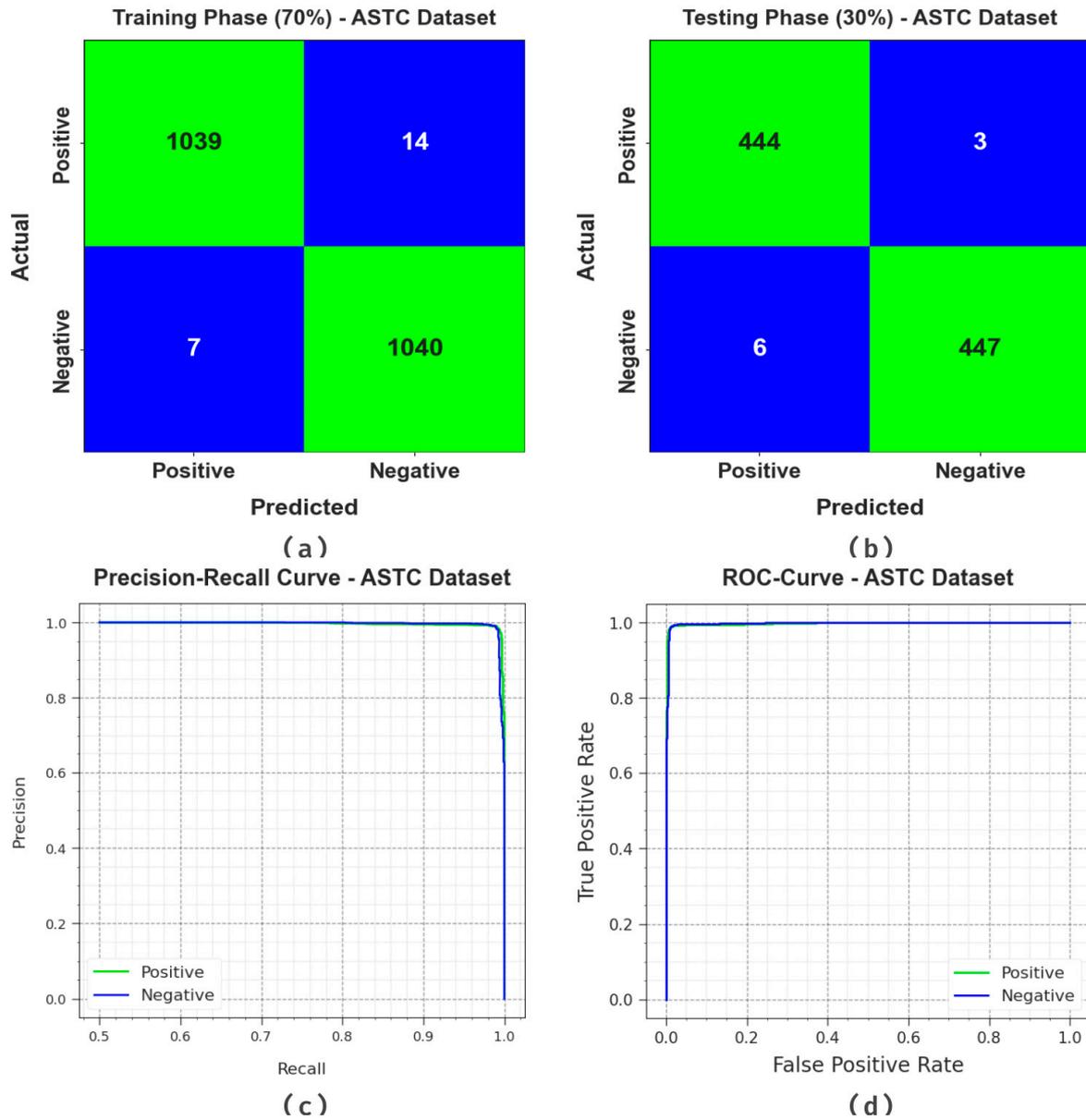


Figure 8. ASTC database: (a,b) confusion matrices, (c) PR_curve, and (d) ROC.

Table 3. Classifier outcomes derived from using the LRLP-IDLHPO algorithm on the ASTC database.

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{Score}	$G_{Measure}$
TR set (70%)					
Positive	98.67	99.33	98.67	99.00	99.00
Negative	99.33	98.67	99.33	99.00	99.00
Average	99.00	99.00	99.00	99.00	99.00
TS set (30%)					
Positive	99.33	98.67	99.33	99.00	99.00
Negative	98.68	99.33	98.68	99.00	99.00
Average	99.00	99.00	99.00	99.00	99.00

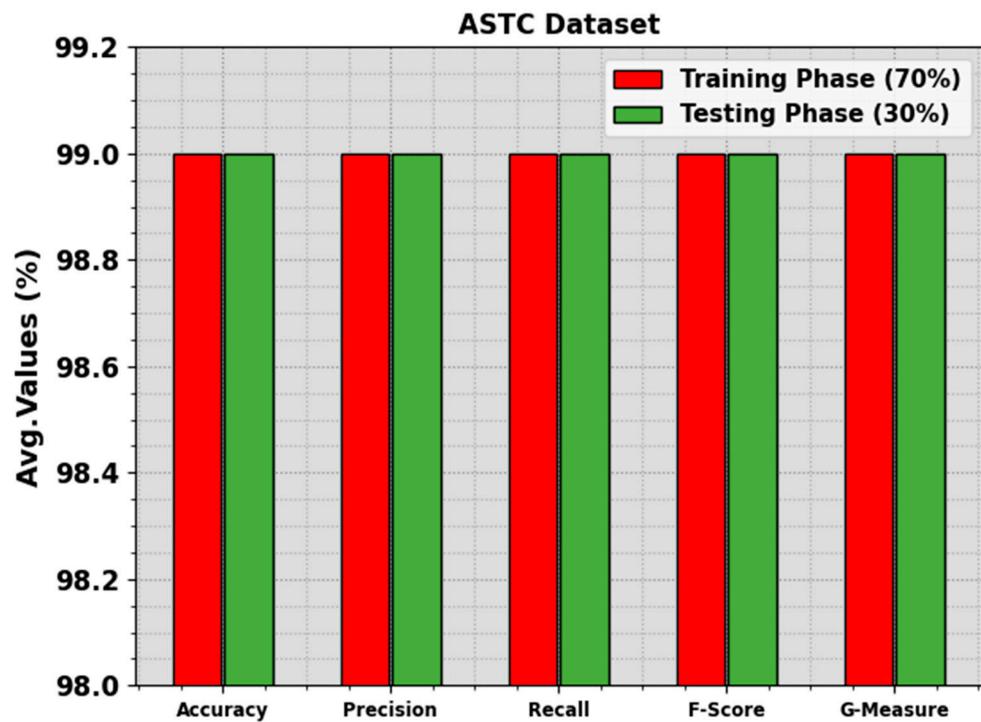


Figure 9. Average values derived from using the LRLP-IDLHPO algorithm on the ASTC database.

Figure 10 illustrates the training accuracy (TR_accu_y) and validation accuracy (VL_accu_y) curves derived from using the LRLP-IDLHPO algorithm on the ASTC database. The TR_accu_y is determined by the estimation of the LRLP-IDLHPO system on the TR database, whereas the VL_accu_y is calculated by evaluating the performance on a separate testing database. The outcomes revealed that TR_accu_y and VL_accu_y rise with an increase in epochs. Thus, the performance of the LRLP-IDLHPO approach improves when used on the TR and TS databases with an increase in the number of epochs.

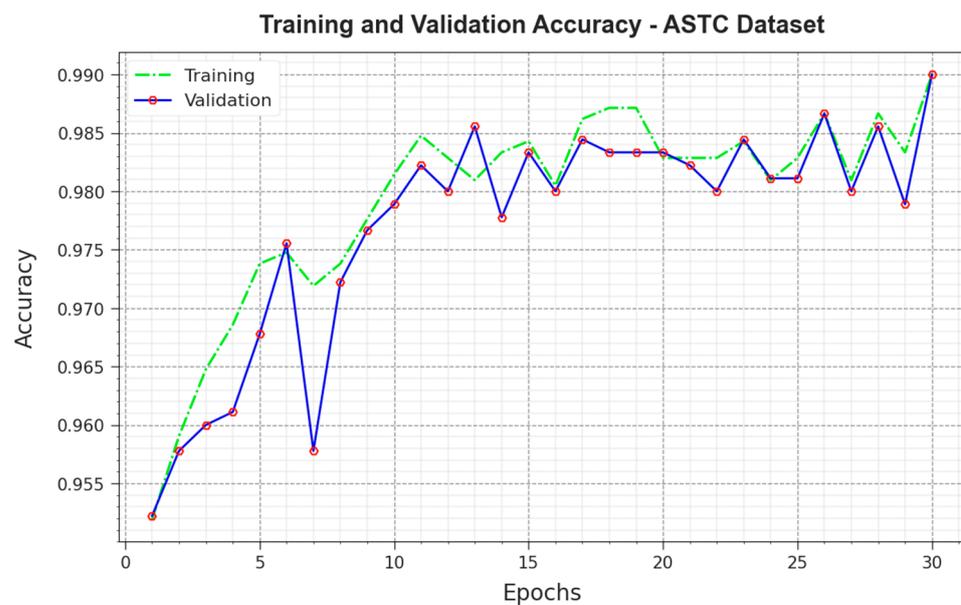


Figure 10. $Accu_y$ curves derived from using the LRLP-IDLHPO algorithm on the ASTC database.

In Figure 11, the TR_loss and VR_loss curves derived from using the LRLP-IDLHPO method on the ASTC database are shown. The TR_loss defines the error between the predictive outcome and original values on the TR data. The VR_loss measures the performance of

the LRLP-IDLHPO algorithm on individual validation data points. The outcomes indicate that the TR_loss and VR_loss tend to reduce with increasing epochs. They also indicate the improved performance of the LRLP-IDLHPO system and its ability to generate accurate classification. The decreased TR_loss and VR_loss values suggest the superior performance of the LRLP-IDLHPO approach in terms of capturing relationships and patterns.

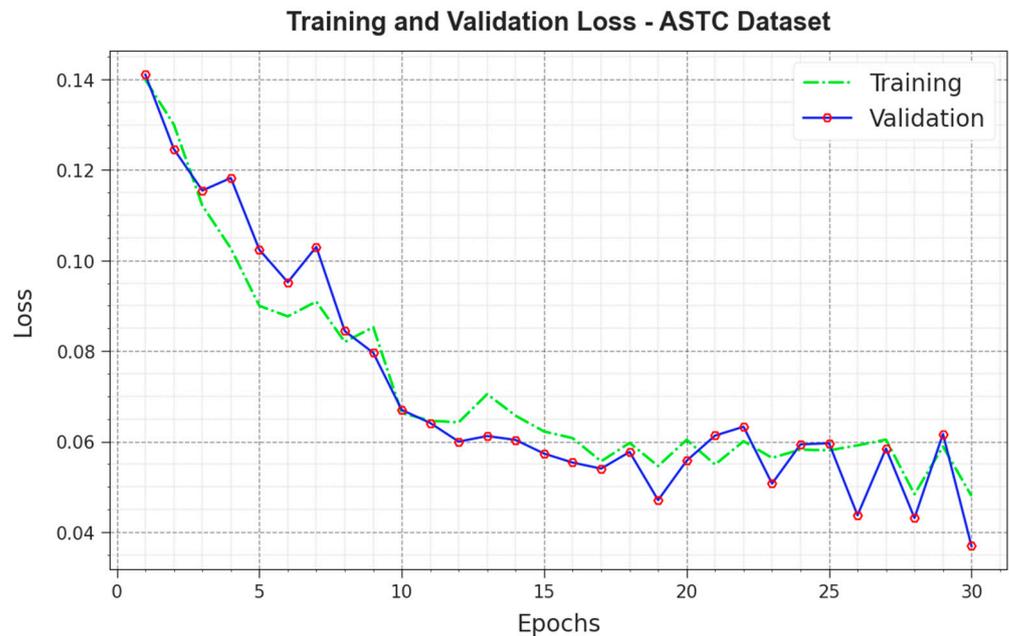


Figure 11. $Accu_y$ loss curves of LRLP-IDLHPO algorithm on ASTC database.

A comparison of the values obtained via using the LRLP-IDLHPO technique and other similar models on the ASTC databases are stated in Table 4 and Figure 12 [25–27]. The outcomes specify that the NB model obtains worse results, whereas the DT and LR techniques exhibit performances closer to that achieved by the LRLP-IDLHPO technique. Additionally, the RNN and GRU systems exhibit reasonable performances. Although the LSTM and IAOADL-ABSA approaches achieve great outcomes, the LRLP-IDLHPO method shows superior outcomes, with higher $accu_y$, $prec_n$, $reca_1$, and F_{score} values of 99%, 99%, 99%, and 99% respectively.

Table 4. Comparison of the results derived from using the LRLP-IDLHPO system and other methodologies on the ASTC database.

Approach Models	ASTC Database			
	$Accu_y$	$Prec_n$	$Reca_1$	F_{Score}
Decision Tree [24]	91.40	90.45	89.78	92.59
Logistic Regression [24]	92.97	92.51	92.42	91.85
Naive Bayes [24]	86.76	87.11	87.42	87.62
RNN [25]	94.67	96.90	96.96	96.40
LSTM [25]	97.96	97.86	95.73	98.29
GRU [25]	95.61	96.46	95.92	96.42
IAOADL_ABSA [26]	98.89	98.88	98.89	98.87
LRLP-IDLHPO	99.00	99.00	99.00	99.00

Thus, the results suggest that the LRLP-IDLHPO technique is an accurate tool for sentiment classification. The LRLP-IDLHPO method achieves better performance over existing approaches through a combination of innovative strategies tailored to the challenges of LRL SA. By incorporating data preprocessing to improve data usability, leveraging advanced word embeddings with SentiBERT, employing EWAG-GRU for effective sentiment classification, and fine-tuning model parameters with HPO, this technique addresses the

key challenges posed by LRLs. The meticulous design of the LRLP-IDLHPO technique enhances each step of the SA pipeline, resulting in better robustness and accuracy, making it suitable to the unique linguistic characteristics and resource constraints of LRLs. Comprehensive simulation analyses validated these advancements, underscoring the method's superiority and reinforcing its potential as a transformative solution in the realm of LRL processing.

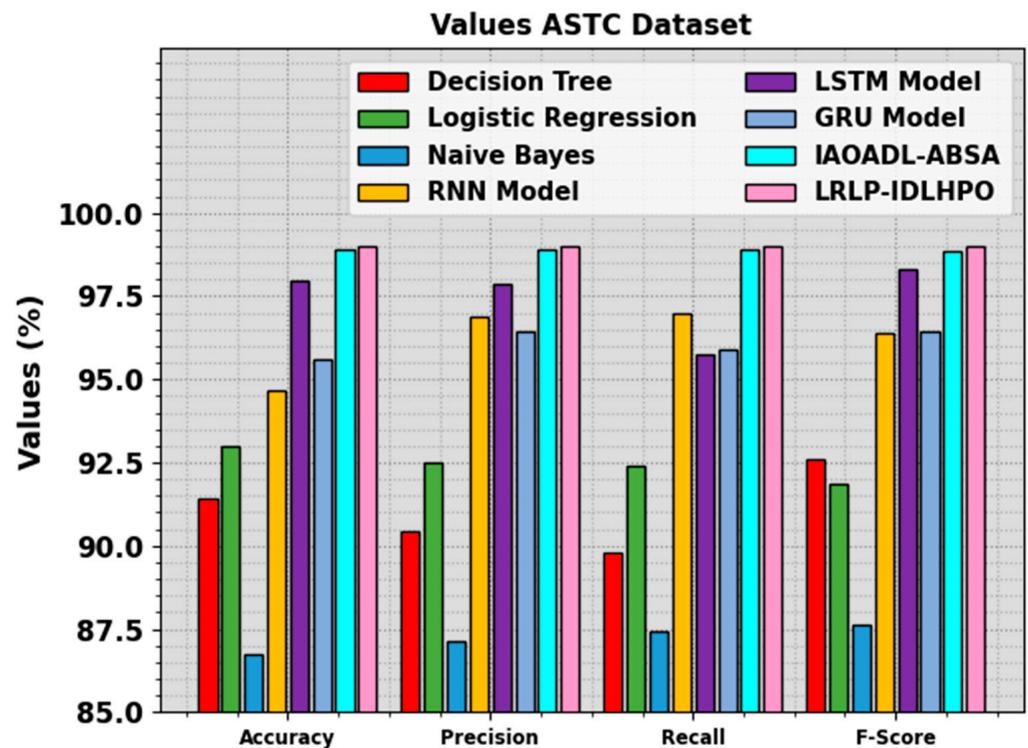


Figure 12. Graphical comparison of the results derived from using the LRLP-IDLHPO system and other methodologies on the ASTC database.

5. Conclusions

In this manuscript, we have proposed the use of the LRLP-IDLHPO approach for the processing of LRLs. The LRLP-IDLHPO technique enables the detection and classification of the different kinds of sentiments present in LRL data. To accomplish this, the presented LRLP-IDLHPO technique incorporates pre-processing, SentBERT, the EWAG-GRU model, and the HPO algorithm for hyperparameter tuning. The EWAG-GRU model is an enhanced RNN that has the capability of processing temporal features with the inclusion of an attention strategy. Finally, the performance of the EWAG-GRU model can be boosted via the addition of the HPO algorithm for the hyperparameter tuning process. A widespread simulation analysis was performed to validate the superior performance of the LRLP-IDLHPO system. The extensive results indicate that the performance of the LRLP-IDLHPO technique is significantly superior compared to the state-of-the-art approaches described in the literature.

Author Contributions: Conceptualization, F.N.A.-W.; methodology, F.N.A.-W. and H.J.A.; software, A.E.O.; validation, A.E.O.; investigation, H.J.A.; data curation, A.E.O.; writing—original draft, F.N.A.-W., H.J.A. and E.S.A.E.; writing—review and editing, F.N.A.-W., H.J.A., A.E.O. and E.S.A.E.; visualization, A.E.O.; project administration, F.N.A.-W.; funding acquisition, F.N.A.-W. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through a large group Research Project under grant number (RGP2/10/44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R281), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444).

Data Availability Statement: The dataset source is given in the article.

Conflicts of Interest: The authors declare no conflict of interest. The manuscript was written through the collaboration between all authors. All authors have approved the final version of the manuscript.

References

1. Keinan, R.; HaCohen-Kerner, Y. JCT at SemEval-2023 Tasks 12 A and 12B: Sentiment Analysis for Tweets Written in Low-resource African Languages using Various Machine Learning and Deep Learning Methods, Resampling, and HyperParameter Tuning. In Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Toronto, ON, Canada, 9–14 July 2023; pp. 365–378.
2. Kokab, S.T.; Asghar, S.; Naz, S. Transformer-based deep learning models for the sentiment analysis of social media data. *Array* **2022**, *14*, 100157. [[CrossRef](#)]
3. Raychawdhary, N.; Das, A.; Dozier, G.; Seals, C.D. Seals_Lab at SemEval-2023 Task 12: Sentiment Analysis for Low-resource African Languages, Hausa and Igbo. In Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023), Toronto, ON, Canada, 9–14 July 2023; pp. 1508–1517.
4. Chang, Y.C.; Ku, C.H.; Le Nguyen, D.D. Predicting aspect-based sentiment using deep learning and information visualization: The impact of COVID-19 on the airline industry. *Inf. Manag.* **2022**, *59*, 103587. [[CrossRef](#)]
5. Bashir, M.F.; Javed, A.R.; Arshad, M.U.; Gadekallu, T.R.; Shahzad, W.; Beg, M.O. Context-aware Emotion Detection from Low-resource Urdu Language Using Deep Neural Network. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**, *22*, 1–30. [[CrossRef](#)]
6. Khan, L.; Amjad, A.; Ashraf, N.; Chang, H.T. Multi-class sentiment analysis of Urdu text using multilingual BERT. *Sci. Rep.* **2022**, *12*, 5436. [[CrossRef](#)] [[PubMed](#)]
7. Dong, J. Natural Language Processing Pretraining Language Model for Computer Intelligent Recognition Technology. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**. [[CrossRef](#)]
8. Yusup, A.; Chen, D.; Ge, Y.; Mao, H.; Wang, N. Resource Construction and Ensemble Learning-based Sentiment Analysis for the Low-resource Language Uyghur. *J. Internet Technol.* **2023**, *24*, 1009–1016. [[CrossRef](#)]
9. Agüero-Torales, M.M.; López-Herrera, A.G.; Vilares, D. Multidimensional affective analysis for low-resource languages: A use case with guarani-spanish code-switching language. *Cogn. Comput.* **2023**, *15*, 1391–1406. [[CrossRef](#)]
10. Kamyab, M.; Liu, G.; Rasool, A.; Adjeisah, M. ACR-SA: Attention-based deep model through two-channel CNN and Bi-RNN for sentiment analysis. *PeerJ Comput. Sci.* **2022**, *8*, e877. [[CrossRef](#)]
11. William, P.; Shrivastava, A.; Chauhan, P.S.; Raja, M.; Ojha, S.B.; Kumar, K. Natural Language processing implementation for sentiment analysis on tweets. In *Mobile Radio Communications and 5G Networks: Proceedings of Third MRCN 2022*; Springer Nature: Singapore, 2023; pp. 317–327.
12. AlBadani, B.; Shi, R.; Dong, J. A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM. *Appl. Syst. Innov.* **2022**, *5*, 13. [[CrossRef](#)]
13. Anand, M.; Sahay, K.B.; Ahmed, M.A.; Sultan, D.; Chandan, R.R.; Singh, B. Deep learning and natural language processing in computation for offensive language detection in online social networks by feature selection and ensemble classification techniques. *Theor. Comput. Sci.* **2023**, *943*, 203–218. [[CrossRef](#)]
14. Kumar, V.S.; Alemran, A.; Karras, D.A.; Gupta, S.K.; Dixit, C.K.; Haralayya, B. Natural Language Processing using Graph Neural Network for Text Classification. In Proceedings of the 2022 International Conference on Knowledge Engineering and Communication Systems (ICKES), Chickballapur, India, 28–29 December 2022; pp. 1–5.
15. Divyapushpalakshmi, M. Ramalakshmi, RAN efficient sentimental analysis using hybrid deep learning and optimization technique for Twitter using parts of speech (POS) tagging. *Int. J. Speech Technol.* **2021**, *24*, 329–339. [[CrossRef](#)]
16. Alyoubi, K.H.; Sharma, A. A Deep CRNN-Based Sentiment Analysis System with Hybrid BERT Embedding. *Int. J. Pattern Recognit. Artif. Intell.* **2023**, *37*, 2352006. [[CrossRef](#)]
17. Rodrigues, A.P.; Fernandes, R.; Shetty, A.; Lakshmana, K.; Shafi, R.M. Real-time twitter spam detection and sentiment analysis using machine learning and deep learning techniques. *Comput. Intell. Neurosci.* **2022**, *2022*, 5211949. [[CrossRef](#)] [[PubMed](#)]
18. Zuheros, C.; Martínez-Cámara, E.; Herrera-Viedma, E.; Herrera, F. Sentiment analysis based multi-person multi-criteria decision making methodology using natural language processing and deep learning for smarter decision aid. Case study of restaurant choice using TripAdvisor reviews. *Inf. Fusion* **2021**, *68*, 22–36. [[CrossRef](#)]
19. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.

20. Huang, X.; Yuan, Y.; Chang, C.; Gao, Y.; Zheng, C.; Yan, L. Human Activity Recognition Method Based on Edge Computing-Assisted and GRU Deep Learning Network. *Appl. Sci.* **2023**, *13*, 9059. [[CrossRef](#)]
21. AbdelAty, A.M.; Yousri, D.; Chelloug, S.; Alduailij, M.; Abd Elaziz, M. Fractional order adaptive hunter-prey optimizer for feature selection. *Alex. Eng. J.* **2023**, *75*, 531–547. [[CrossRef](#)]
22. Akhtar, M.S.; Ekbal, A.; Bhattacharyya, P. Aspect based sentiment analysis: Category detection and sentiment classification for Hindi. In *Computational Linguistics and Intelligent Text Processing, CICLing 2016*; Gelbukh, A., Ed.; Springer: Cham, Switzerland, 2016; pp. 246–257.
23. Arabic Sentiment Twitter Corpus. Available online: <https://www.kaggle.com/mksaad/arabic-sentiment-twitter-corpus> (accessed on 2 April 2023).
24. De Diego, I.M.; Redondo, A.R.; Fernández, R.R.; Navarro, J.; Moguerza, J.M. General Performance Score for classification problems. *Appl. Intell.* **2022**, *52*, 12049–12063. [[CrossRef](#)]
25. Pathak, A.; Kumar, S.; Roy, P.P.; Kim, B.G. Aspect-based sentiment analysis in Hindi language by ensembling pre-trained mBERT models. *Electronics* **2021**, *10*, 2641. [[CrossRef](#)]
26. Saleh, H.; Mostafa, S.; Alharbi, A.; El-Sappagh, S.; Alkhalifah, T. Heterogeneous ensemble deep learning model for enhanced Arabic sentiment analysis. *Sensors* **2022**, *22*, 3707. [[CrossRef](#)]
27. Rasool, H.A.; Abedi, F.; Ismaeel, A.G.; Abbas, A.H.; Khalid, R.; Alkhayyat, A.; Jaber, M.M.; Garg, A. Pelican Optimization Algorithm with Deep Learning for Aspect based Sentiment Analysis on Asian Low Resource Languages. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.