RSS-LSTM: A Metaheuristic-Driven Optimization Approach for Efficient Text Classification

Muhammad Nasir[1], Noor Azah Samsudin[2], Shamsul Kamal Ahmad Khalid[3],
Souad Baowidan[4]*, Dr. Humaira Arshad[5], Wareesa Sharif [6]
Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn (UTHM), Johor Bahru, Malaysia[1, 2, 3]
Faculty of Computing and IT, King Abdulaziz University, Jeddah, Saudi Arabia[4]
Department of Computer Science, The Islamia University of Bahawalpur (The IUB), Bahawalpur, Pakistan[5]
Department of Artificial Intelligence, The Islamia University of Bahawalpur (The IUB), Bahawalpur, Pakistan[6]

***Abstract*—The digital data consumed by the average user daily is huge now and is increasing daily all over the world, which requires sophisticated methods to automatically process data, such as retrieving, searching, and formatting the data, particularly for classifying text data. Long Short-Term Memory (LSTM) is a prominent deep learning model for text classification. Several metaheuristic approaches, such as the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Firefly Algorithm (FF), have also been used to optimize Deep Learning (DL) models for classification. This study introduced an improved technique for text classification, called RSS-LSTM. The proposed technique optimized the hyperparameters and kernel function of LSTM through the Ringed Seal Search (RSS) algorithm to enhance simplification and learning ability. This work was also compared and evaluated against state-of-the-art techniques such as GA-LSTM, PSO-LSTM, and FF-LSTM. The results showed significantly better results using the proposed techniques, with an accuracy of 96%, recall of 96%, precision of 96%, and 95% f-measure on the Reuters-21578 dataset. In addition, it showed an accuracy of 77%, recall of 77%, precision of 78%, and f-measure of 76% on the 20 Newsgroups dataset, while it achieved accuracy, recall, precision, and f-measure of 91%, 91%, 94%, and 90%, respectively, using the AG News dataset.**

Keywords*—*Deep learning; text classification; Long Short-Term Memory; Ringed Seal Search; metaheuristic algorithms; Part Swarm Optimization; Genetic Algorithm; Firefly Algorithm; hyperparameter optimization

I. INTRODUCTION

A large amount of textual data and different types of content are distributed to millions of people worldwide on the internet. The significant increase in the size of online data has attracted significant attention nowadays. Because of the large increase in textual data worldwide, the demand for text classification has also increased [1]. Hence, searching for a specific document within a large collection has become a difficult challenge. It examines people's emotions and distinguishes between customer comments on specific topics. Text classification (text categorization or tagging) involves assigning a text document to a set of predefined labels or classes using different machine learning and deep learning methods [2]. Different text classification techniques have been widely employed to categorize and organize content. It is important to gather and categorize documents automatically, according to their content. The primary objective of text classification is to divide unstructured documents into appropriate groups according to their content [3].

Machine Learning (ML) and Artificial Intelligence (AI) have gained significant prominence in recent years, emerging as highly discussed subjects. Numerous machine learning methods have achieved remarkable results in Natural Language Processing (NLP) [4]. However, conventional machine-learning-based text classification techniques have several drawbacks, including dimension explosion, data sparsity, and generalization capacity and selecting the optimal parameters for models. Most earlier research must consider the possibility that data could be misplaced or misconstrued following neural network computations [5]. The development of new machine learning techniques has yielded significant advancements in recent years, and deep learning has received more attention in the context of text categorization [6]. Deep learning techniques have been successful in the past few years, and there has been a significant increase in studies in this field, signifying that deep learning approaches have outperformed traditional machine-learning-based approaches in several text classification tasks, such as sentiment analysis, news categorization, question answering, and natural language inference [7].

Natural Language Processing has prioritized text classification for a long time. Currently, methodologies and techniques constitute integral components of numerous products and are deemed essential for a wide array of applications and devices. Many deep learning architectures, including LSTM Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), and more recent transformers, have been applied to attain various state-of-the-art outcomes in NLP tasks [8]. Text classification is a major process in Natural Language Processing, and recent research has focused on deep learning-based neural network techniques that have shown promise. However, previous studies have often overlooked the potential loss or misinterpretation of information in neural network calculations. A study introduced LSTM-Com, a technique that leverages historical information, such as the original text and hidden layer outputs, to address these issues. LSTM-Com dynamically selects important historical information to compensate for the neural network, resulting in improved performance compared with the baseline in the classification experiments. A Long Short-Term Memory approach overcomes these challenges by performing text classification using historical data, including the original text data and output data

*Corresponding Author.

from the hidden layers [9]. Deep learning-based techniques are more beneficial for text classification compared to machine learning that uses traditional text classifiers, which have flaws such as data sparsity, dimension proliferation, and less generality; instead, the classifiers based on deep learning techniques can magnificently overcome these defects because they have strong learning ability with higher prediction accuracy; in contrast, they also avoid a cumbersome feature extraction process [10].

To build an LSTM model for classification, machine learning researchers manually configure certain parameters that are independent of the data. These parameters, such as the network structure and the training process of the LSTM, are known as hyperparameters. A key challenge is to find a set of hyperparameters that yield an accurate model within a reasonable timeframe, which is an integral part of the hyperparameter optimization problem [11]. The optimization of hyperparameters plays an important role in the performance of machine-learning algorithms. The importance of hyperparameter optimization is well recognized; however, there has been limited research to confirm its assumptions. Hyperparameter optimization is crucial for deep learning models because it directly affects their performance and generalization. Without optimal hyperparameters, a model may underfit or overfit, failing to capture the underlying patterns in the data. Tuning helps strike the right balance, maximizing the model's performance on unseen data [12]. Hyperparameter optimization is critical in machine learning. Machine-learning algorithms require the setting of hyperparameters before training the model. These values significantly impact the model's performance, but finding good ones is complex, which has led machine learning researchers to look into automated methods for hyperparameter searches [13].

A previous study showed that a meta-heuristics-based algorithm has a significant effect on optimization. In addition, when an optimization problem's dimension (the number of routes or furnished) is increased, a study of the Grasshopper Optimization Algorithm (GOA) explains it [14]. The hyperparameters of a CNN significantly affect its performance and adjusting them manually is laborious and ineffective. The Spotted Hyena Optimization (SHO) is a high-level metaheuristic optimization algorithm with sophisticated exploration and exploitation capabilities. SHO produces a set of solutions in the form of hyperparameters that must be tuned. This procedure was repeated until the target optimal solution was attained [15].

Balancing exploration and exploitation play a central role in defining the effectiveness of an evolutionary algorithm. Optimal performance necessitates varying levels of the exploration-exploitation trade-off at different stages of evolution [16]. A study discussed the importance of balancing local exploitation and global exploration in metaheuristic algorithms and elaborated on the effectiveness of the bat algorithm in achieving this balance. Comparing the bat algorithm with the recurrent search approach, the study demonstrates that the bat algorithm is superior; it also explores the consequences of these findings for higher-dimensional optimization problems and applies the bat algorithm to business and engineering design optimization. A healthy search requires balancing exploration and exploitation [17]. A thorough survey is necessary because various techniques, datasets, and evaluation criteria have been proposed in the literature [18]. Firefly cannot offer a robust mechanism for achieving an ideal balance between exploration and exploitation; it cannot set the parameters strongly [19]. The RSS algorithm is a metaheuristic with two searching states (Brownian and Levy) that substitute randomly because of noise and balance exploitation and exploration of the search, the likelihood of finding local optima quickly is very low. Furthermore, RSS uses significantly fewer parameters than GA, PSO, and FF [20].

The performance of LSTM depends on parameter optimization, which is applied to text classification using various metaheuristic algorithms.

Following major contributions through in proposed approach:

- Pre-processing of data to make it efficient for further processing for feature engineering and deep learning model.
- An enhanced LSTM with RSS is proposed for textual data classification. Hyperparameters have a significant impact on the performance of deep learning models. With this measure of data, choosing the appropriate parameters (by optimizing the kernel function) for a neural network has become a huge exploration region in recent research.
- The proposed technique is compared with existing techniques, such as GA, PSO, and FF, using LSTM. To validate the performance of the proposed model for four measuring matrices: Accuracy, Precision, Recall, and F1-Score.
- Evaluate the existing technique and the proposed technique using three benchmark datasets: 20 Newsgroups, Reuters-21578, and AG News.

Section II explains literature, Section III explains the proposed approach, and Section IV explains the results of the experiments and Section V elaborates the conclusion of this study.

II. LITERATURE REVIEW

Text classification applications include spam filtering, contextual search, opinion mining, product review analysis, content management, and text sentiment mining. When tuning the hyperparameter automatically using an algorithm, it is called auto-tuning, which offers an effective way to automatically train the process of a model, although it provides more efficient results [21].

Several studies have described different machine learning and deep learning models, including pre-processing of text classification, related calculations, and test methods [22-24]. A minimalist and multi-propose-based text classification approach, uTC, was tested on 30 different datasets and showed the best accuracy compared with other state-of-the-art classification methods [23]. Another study suggested a method for determining sentiment review comparisons using three feature extraction techniques: Word2vec, Doc2vec, and TF-

IDF. It uses machine learning algorithms, such as SVM, Naive Bayes, and Decision Tree, with a grid search for optimization. The performance of these algorithms was assessed based on accuracy [24]. The advantages and disadvantages of related models are sorted, and the CNN model can capture the important content of the text. By contrast, the RNN model can analyze the context. The deep learning method is applied to text classification, which saves a large amount of workforce and material resources and improves the accuracy of text classification [25].

A sampling technique was proposed to solve the imbalanced class distribution for classifying tweets using Random Forest, Naive Bayes, and XGBoost [26]. In text classification tasks, the long short-term memory network and convolutional neural network models can both achieve high classification accuracy, and different deep learning models propose feature engineering using Term Frequency-inverse Document Frequency (TF-IDF) and also compare it with CNN, LSTM, and LSTM Attention for short- and long-text classification [27]. A hybrid model using CNN and LSTM was proposed for text classification, in which the features of text sentences were extracted using a multi-scale CNN, and the dependence of the text context was then captured using an LSTM model [28]. A supervised weighing scheme called the term frequency-inverse category frequency model proposed for text classification using deep learning was proposed for five different datasets this research covers to overcome the computational cost compared to other deep learning models [29].

Although resource-intensive, hyperparameter optimization is essential for machine learning. A novel technique called AgentHPO, which automates this procedure, examines task data on its own, experiments with different hyperparameters, and optimizes them based on past performance. Compared to conventional methods, this methodology streamlines the setup, lowers the number of trials required, and improves interpretability. Empirical tests reveal that AgentHPO frequently performs better than human trials and yields interpretable outcomes [30].

Multiple hyperparameters need to be set and tuned for the deep learning model's evaluation to predict the early onset of Parkinson's disease using hyperparameter optimization of the deep learning model [31]. The MLearn-ATC algorithm was compared with popular algorithms for classification, including Support Vector Machines, Probabilistic Neural Networks (PNN), K-Nearest Neighbor (KNN), and Naïve Bayes [32], to solve the text categorization issue, they proposed approach was examined using three distinct document datasets: Reuters-21578, 20 Newsgroups, and Real dataset [33].

Comparative investigations on the various feature selection and classification techniques used in sentiment analysis based on Natural Language Processing and contemporary techniques such as the Genetic Algorithm and rough set theory are evaluated. Another study examined the differences between sentiment analysis and standard feature-selection techniques for text categorization [34]. Several studies using supervised and unsupervised learning methods have been conducted to solve the issue of fake news identification. A study was conducted using the ISOT dataset to identify fake news. Long Short-Term Memory is applied in the developed model to distinguish between fake and authentic news, and hyperparameter tuning techniques, including grid search and random search, are proposed to adjust the model's hyper-parameters [35].

A role-based access control (RBAC) strategy is required to precisely identify access permissions to secure data. SQL queries created by authorized users have extremely similar features and are challenging to separate. A CNN-LSTM based on Part Swarm Optimization was proposed for hyperparameter optimization to detect attacks in SQL queries. Stock market uncertainty has a significant impact on many global financial and economic activities. Setting an investment plan or choosing the best time to trade depends on forecasting stock price movement, and a PSO-LSTM-based technique was proposed for stock price forecasting [36]. Convolution layer and Bidirectional LSTM (BiLSTM) with the attention mechanism proposed for text classification, particularly sentiment analysis. However, there is still an issue that LSTM cannot distinguish the different relevance between each part of the document [37].

A cross-entropy trained-based deep learning model called a bidirectional LSTM network is employed to perform text classification, utilizing both supervised and semi-supervised procedures, and an evaluation test using IMDB and AG News Group datasets [38]. The CNN-LSTM-based NC2LO Caledonian crow-optimization-based hybrid approach was applied for short-text classification. It was applied to IMDb, Tagmy News, Twitter, and AG News datasets, for developing tool modeling skills and attainment attracted, this Caledonian crow optimization model employs both social and asocial learning [39]. Another study reviewed metaheuristic optimization algorithms for power systems, focusing on their ability to solve complex optimization problems in environments with limited information and computational resources. It discusses six key challenges in power systems and evaluates the effectiveness of various metaheuristic algorithms in addressing them, evaluating their importance in promoting environmental sustainability and supporting renewable energy sources. The effectiveness of a metaheuristic algorithm is mostly determined by how well it balances globally diversified exploration and local intensive exploitation [40].

A hybrid technique using bidirectional long short-term memory (BiLSTM) and bidirectional encoder representations from transformers (BERT)-based approach was proposed for text mining to understand Chinese railway incidents caused by electromagnetic interference. A text mining technique using TextBlob for sentiment score with TF-IDF vectorization and a Linear SVC classification model was proposed for text mining the Covid-19 vaccination Twitter dataset [41]. Table I gives a comprehensive review of related work.

Ship pilots must have a thorough understanding of the future positions of their ships and their target ship at a given time. However, there are now important problems that need to be resolved regarding forecast accuracy and computing efficiency. The deep, long, short-term memory network architecture and genetic algorithm were developed in this study to address these issues and predict the shipping route of inland water. The GA-LSTM model effectively increased the precision and speed of trajectory prediction [56]. Convolutional neural networks

(CNNs) have gained recognition for their promising performance in text categorization and sentiment analysis because they can preserve a document's 1D spatial orientation, where the order of words is crucial. Research has been conducted using genetic algorithms to automatically determine the ideal network architecture without the need for any intervention from experts [57]. Researchers use machine learning, and some use deep learning models to solve the classification issue, and Artificial Neural Networks (ANN), which are implemented with GA for text classification, although ANN also has the main problem of tuning hyperparameters [53].

A study investigated the impact of LSTM parameter optimization with meteoritic algorithms on text classification performance. The GA-LSTM automatically chooses settings to create the best gene subset. The original Cuckoo Search [58] optimization enters the local optimum because of the high dimensions of the early convergence of complicated issues. CS, PSO, and GA dominate global optimization algorithms in scientific and technological applications. These algorithms have limitations in the development of novel solutions to preserve the equilibrium between exploration and exploitation [59]. Choosing the best hyperparameter of a model has an immediate effect on the model's performance, and another study showed that the Bayesian Optimization [60] technique is a more viable technique for the performance of the K-Nearest Neighbor model than other models [61].

TABLE I. LITERATURE REVIEW

Year	Ref.	Technique	Limitations	Outcome(s)	Dataset
2024	[42]	CNN-Bi-LSTM	Improvements could involve incorporating additional variables	CNN-Bi-LSTM model adapts and achieves coefficients of determination, RMSE, and RMAE of 0.95, 37.94, and 5.27, respectively.	Gold prices dataset
2024	[43]	GS-CNN-LSTM	Probable overfitting of the model. Model Complexity. The test model generalizes well to unseen data	Hybrid model with grid search achieves 91.67% accuracy, 89.66% recall, 93.55% specificity, 92.86% precision, 91.23% f1-score, and 0.9310 AUC.	Heart disease Cleveland
2023	[44]	CNN-LSTM	Limited comparison. Hybridization models complexity	In this study hybrid model achieves an accuracy of 93.51%, outperforming traditional ML models in detecting PD using dynamic features.	PC-GITA disease
2023	[45]	LSTM-RNN-GRU	Economic indicators, humidity, and seasonal factors could also significantly impact electrical load that is not considered here.	In this study, Deep learning models, including LSTM, GRU, and RNN, are used for load forecasting. GRU model achieves the best performance.	Forecast electricity load in Palestine based on a novel real dataset
2022	[46]	LSTM-AE-TPE	Model complexity. Computational Cost.	The proposed model in this study achieves an R-square of over 0.9, indicating its effectiveness in indoor temperature prediction	Temperature dataset
2021	[47]	GA- Deep Long Short-Term Memory	Increased computation time. Increased complexity.	DLSTM model that achieves RMSE using Dynamic-Adam as of 0.026 and using Dynamic-Adamax 0.006	Power load dataset
2021	[48]	BO-PSO-RNN	Model complexity concerns. Limited comparison. The model has not yet been tested in high-dimensional spaces.	In this study RNN and LSTM models demonstrate their effectiveness compared to other methods like BO-L-BFGS-B and BO-TNC	Stock market price data. Oilfield production
2021	[49]	Hyperparameter Exploration LSTM-Predictor (HELP-LSTM)	Sequence length and number of fully connected layer units can impact performance. The HELP algorithm might experience collapse due to its extreme hyperparameter	This study utilizes probability-based exploration with LSTM-based prediction to improve hyperparameter exploration in neural network training.	MNIST
2021	[50]	GA-DNN	Computational cost. Kind of black-box mature. Model complexity	GA-based approach achieves 75.86% for RNNs and 41.12% for DNNs.	Sample streaming-data, Indian stock market, MNIST, CIFAR10
2020	[51]	PSO-LSTM, PSO-ANN	Limited country level dataset is used, and validation of dataset may also require	The PSO-LSTM model improves prediction accuracy and stability for water level forecasting compared to ANN. It enhances flood prediction at varying lead times, aiding future flood risk mitigation efforts in the study region.	Watersheds dataset used in this study
2020	[52]	GWO-LSTM	Address only global optima. Limited comparison.	The Grey Wolf Optimizer (GWO) algorithm to optimize the hyperparameters of Long Short-Term Memory models for language modeling tasks performed in this study.	Penn treebank dataset
2019	[53]	GA-LSTM	Computational cost. Performance issues for finding the global optimum	GA-LSTM optimization technique and achieve a maximum of 55% accuracy.	
2018	[54]	Differential Evolution-LSTM	DE algorithm took more processing time	LSTM hyperparameters improve emotion recognition. The proposed framework achieved 77.68% accuracy.	the dataset collected from wireless wearable sensors (Emotive and Expatica E4)
2018	[55]	CSA-LSTM	The hybrid model has not been evaluated on various datasets containing different text types.	Competitive search algorithm (CSA) used with LSTM and shows higher results.	Reuters-21578, RCV1-v2 and EUR-Lex

III. PROPOSED APPROACH

An improved text classification technique is proposed using Ringed Seal Search, Long Short-Term Memory and the model named RSS-LSTM, which demonstrated a considerable impact of LSTM hyperparameter optimization with RSS by optimization of the kernel. The RSS-LSTM technique proposed herein achieves a balanced approach between exploitation and exploration. The proposed research suggests a strong technique for hyperparameter optimization for text classification that produces more optimized results than the existing methods. This study explicitly compared using 20 Newsgroups, Reuters-21578, and AG News datasets for multiple labeled text classification. In this study, three datasets were taken from the Kaggle and UC Irvin machine learning repositories: 20 Newsgroups, Reuters-21578, and AG News. The datasets were used to evaluate the effectiveness of the RSS-LSTM. The experiment was programmed using the Jupyter Notebook for the proposed RSS-LSTM, at HP Xeone Workstation z440, 32gb RAM and 2.4 processor. The datasets used in this experiment were selected based on the extent to which they liked. There are two parts to the datasets; 30% of the data were chosen for testing, while 70% were used for training.

Fig. 1 shows the proposed model stages, which consist of three major stages: Data Pre-processing, Optimization of LSTM parameters, and performance measurement criteria. The details of the three stages are as follows.

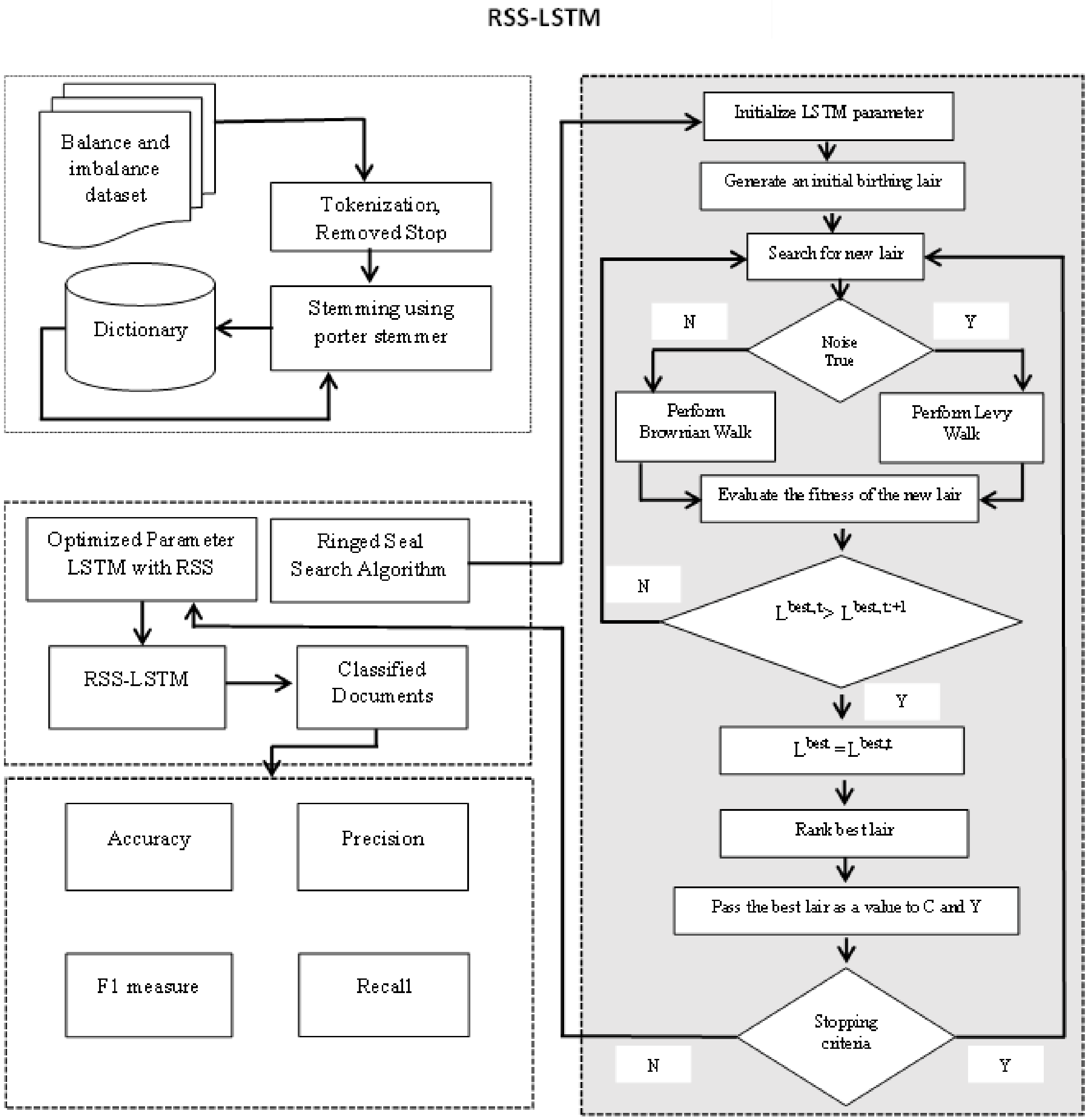

Fig. 1. Proposed model RSS-LSTM.

A. Stage 01

Pre-Processing: In this stage 01 textual dataset, such as 20 Newsgroups, Reuters-21578, and AG News Group, are pre-processed and used for further experiments. Below list of preprocessing steps performed at this stage:

- Tokenization of datasets and feature extraction.
- Removing spaces and punctuation.
- Removing unnecessary words to proceed with meaningful words.
- Removing emojis and stop words.
- Porter stemmer to remove inflection.

B. Stage 02

Optimization of LSTM Parameters: An enhanced method using RSS based on LSTM was implemented in this stage. The performance of the proposed RSS-LSTM method was measured and compared with existing methods, such as GA-LSTM, PSO-LSTM, and FF-LSTM. The proposed RSS-LSTM showed improved results compared to existing techniques.

C. Stage 03

Performance Measuring: In this stage, the performance of the proposed model is evaluated, which comprises four measuring criteria: accuracy, recall, precession, and F1-score. The results section of the proposed comparison model explains these measurement criteria and their results. The proposed models focus on optimizing deep learning techniques, such as LSTM, to achieve optimal results. Fig. 1 shows the proposed model. Algorithm 1 shows the Pseudo-code of proposed RSS-LSTM technique.

Algorithm 1: Pseudo-code of RSS-LSTM

Start …
1. Set the initial parameter of LSTM
2. Producing starting number of lairs $L_1 = (f = 1,2,3,\ldots,n)$
3. While (Stopping measure)
4. If noise = false
5. Search the nearness for a new layer with a Brownian walk
6. Else
7. Expend the search for a process for a new layer by using levy walk
8. End if
9. Evaluating the fitness of each new lair and comparison with the previous lair
10. If
11. $L^{\text{best,k}} > L^{\text{best,k+1}}$
12. Select the new lair
13. $L^{\text{best}} > L^{\text{best,k}}$
14. Else
15. Go at 4
16. End if
17. Rank the solutions.
18. Return the best lair of execution
19. The global finest lair is fed LSTM classifier for training
20. Training the classifier (LSTM)
21. End while
22. End

D. Proposed Approach RSS-LSTM Explanation

A metaheuristic technique called RSS is suggested to address optimization issues. To escape predators, the RSS method relies on the foundation of seal pup behavior to find the best lair. This technique divides the search space into two states: ordinary, routine, and urgent or fast. Under ordinary and urgent states, an intensive and extensive search is performed to find good-quality air and move in it to escape predators [62].

If it identifies the location of the exploration space where ω = 1 (ω= 0 indicate the normal state), ∂ is notified that **Ω** contains β, a predator that is moving and making noise pointed as ω. A state (**Ω**, ρ) for an E event, where **Ω** is referred to as an urgent state if **Ω** comprises β and ∂ members of the event in the exploration space that has noise ω. Let A be an event and the search space be (**Ω**, β, ∂, ρ).

If the search space is now ρ is ω in a state where ω= 0 (shows the outside noise), then ∂ is assumed to be not informed **Ω**, contains β, and (**Ω**, ρ) represents a normal condition, executes a Levy-walk for an urgent ∂ situation and Brownian-walk for a typical normal condition ∂. The proposed RSS-LSTM is discussed in the next section.

Text Classification: The proposed research developed an enhanced technique using Ringed Seal Search and Long Short-Term Memory. The Hyperparameters of LSTM are optimized using the RSS one of the metaheuristic techniques, which overcomes the efficiency of the other techniques PSO-LSTM, GA-LSTM, and FF-LSTM algorithms. The selection of parameters for the classification problem presents one of the primary obstacles to the optimal LSTM. It is a frequent practice to improve LSTMs by utilizing metaheuristic search algorithms that are inspired by nature to achieve better classification outcomes. A brief description of the LSTM classifier is provided below:

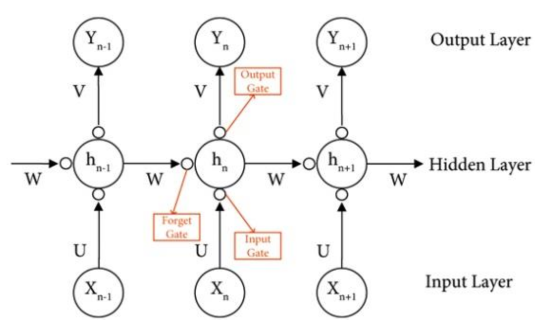

Fig. 2. LSTM Classifier.

As shown Fig. 2 describes the LSTM classifier basic architecture, the input and output data are represented by Xn and Yn, respectively, the weight coefficients are represented by U, V, and W, he is the hidden layer status, and hn is related to the current input Xn input and the previous R hidden layers.

$$h_1 = UX_n + W_{n-1}h_{n-1} + W_{n-2}h_{n-2} + \cdots + W_{n-R}h_{n-R} \quad (1)$$

Finding a hyperplane that appropriately divides the training dataset into two groups and optimizes the LSTM is the main goal. The LSTM classification problem is combined as follows:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$
$$\iota_t = \sigma_g\ (W_i x_t\ +\ U_i\ h_{i-1} + b_i)$$
$$O_t = \sigma_g\ (W_o a_t\ +\ U_o h_{t-1} + b_o) \quad (2)$$
$$c_t\ =\ f_t\ \circ c_{t-1}\ + \iota_t \circ \sigma_c\ (W_c x_t\ +\ U_c h_{t-1} + b_c)$$
$$h_t\ = \sigma_t \circ \sigma_h\ (c_t)$$

Hyperparameters of the error terms were represented by C > 0. The normal vector and offset of the separating hyperplane are the variables denoted by the letters w and b in Eq. (2), respectively. The LSTM parameters were optimized using the Ringed Seal Search. The RSS uses how seal pups search to find the best-hidden place to avoid predators. The proposed RSS algorithm presents a sensitive search paradigm inspired by the movement of the seals. Seal pups are always relocated to high-quality lairs. These lairs offer both thermal shelter against low air temperatures and strong wind chills, as well as shielding themselves from predators, similar to bears. A complex lair can be placed in one place for a seal owing to its close movement. When the seal pup moves throughout its multi-chamber cave and searches for a new one, a sequence of events can be recounted. Evolution was accomplished by altering a random value. The starting population of LSTM parameters is represented by a matrix, the chosen parameters are placed in a vector form, and the vector has evolved to find the optimal combinations of parameters in each iteration.

Inspired by nature, when addressing an optimization problem, the RSS always begins with initial values that can be used as the initial state. The first answer is represented by a vector of values (L_i,i =1, 2,3..k_n.) during the optimization process. The RSS algorithm always begins with a multi-chambered initial number of birthing lairs n. Puppies enter a search area to locate new, better lairs to hide themselves. The formation of an array from these starting values in the search space is required to locate a better search space. Eq. (3) and (4) define the number of lairs in the RSS algorithm that corresponds to the lairs for seal pups. Most metaheuristic algorithms start with the initial population, which can be named as starting values or initial values, because to solve an optimization problem, it is necessary to start with some initial values. Eq 3 represents the initial lair

$$L_i, i = 1,2,3, \ldots, n \quad (3)$$

There are chambers m in every lair, arranged at random. Each L contained m chambers. As an example, consider an array of L = [I multiply m] that represents the lair i of a habitat that is now in use,

$$L = [i \times m] \quad (4)$$

The values range from a predetermined bottom bound, L_bj, to the upper limit, U_bj, randomly and consistently in the search space, as described in Eq. (5).

$$L_i = L_b + (U_b +).rand\ \left(size(L_b)\right) \quad (5)$$
$$i = 1,2,3, \ldots, n \quad (6)$$

where i is the lair number, and n is the number of initialized pairs. The seal travels from one lair to another in a particular search pattern, producing new solutions (new lairs) x t+1 for seal *i*. A new lair is located in the Eq. (7)

$$\chi_i^{t+1}\ = \chi_i^{t+1} + \alpha \times \Delta x \quad (7)$$

where a indicates the size of the step in urgent or normal states.

$$\Delta x =\ {}^{\lambda}levy \quad \text{where}\ w\ =\ 1 \quad (8)$$

where ω represents the uniform discrete distribution shown in Eq. (8), (ω = 1 denotes the external noise). For the Levy walk, the random walk is typified by a step size that is determined using an inverse power-law tail probability distribution, as shown in Eq. (9).

$$Levy \sim u\ \overline{t^{-\lambda}} \quad (9)$$

where t is the length and 1< λ < 3. When λ is less than or equal to 3, there is no heavy tail in the distribution, and the sum of all the lengths approaches a distribution.

Anomalous diffusion, in which the mean squared displacement increases linearly with time, characterizes a Levy walk. The Brownian walk, in contrast to the Levy walk, is typified by normal diffusion, where the mean-squared displacement increases linearly.

Eq. (10) illustrates the structure of the Brownian walk search for a new chamber inside a multichambered lair structure.

$$equal\Delta x =\ {}^{\lambda}brownian \quad \text{where}\ \omega\ =\ 0 \quad (10)$$

The search is characterized by the step size described in Eq. (11).

$$S = K\ \times\ rand(d,\ Ndot) \quad (11)$$

K represents the standard deviation of the regular distribution of the diffusion rate coefficient, d denotes the dimensions of the problem, and N dots symbolize the quantity of Brownian particles within the search space.

The proposed RSS-LSTM approach responds to variations in hyperparameters through its ringed seal search optimization process. This process iteratively explores the hyperparameter space to find the optimal set of hyperparameters that minimizes the objective function, which in this case is the performance of the LSTM model in classification. The RSS algorithm perform a balanced exploration that tries new hyperparameters and exploitation that exploits known good hyperparameters to efficiently search the hyperparameter space for optimal solution.

The Brownian Walk function of the algorithm generates a random walk for a specified step size. Given a current value x in the range of the lower and upper bounds [lower_bound, upper_bound], from the uniform distribution, it adds a random value walk [step_size, step_size] to x. Then, the function checks if the new value [new_value] is within the bounds [lower_bound, upper_bound]; if it exceeds the upper bound then it returns the upper_bound, or if it falls below the lower bound then it returns the lower_bound. Otherwise, it returns to a new value, this function is mathematically represented as:

$$\text{new\_value} = \text{walk} + \text{lower\_bound}$$

where,

$$\text{walk} \sim \text{Uniform}(-\text{step\_size}, \text{step\_size}) \tag{13}$$

The final value is then paired to ensure that it remains within the specified bounds.

Using Pareto distribution, the Levy walk function generates a random walk. First, it samples a value r from a Pareto distribution with the shape parameter beta. It then samples an angle u from the uniform distribution [0, 2/pi] and computes a walk value using r×cos(u). Similar to the Brownian walk, the function ensures that the new value remains within the specified bounds [lower-bound, upper-bound]. Mathematically it can be represented as

$$r \sim \text{Pareto}(\beta)$$

$$u \sim \text{Uniform}(0, 2\pi)$$

$$\text{walk} = r \times \cos(u)$$

$$\text{new\_value} = \text{walk} + \text{lower\_bound} \tag{14}$$

E. Performance Criteria

Accuracy, precision, recall, and f-measure are all important factors to consider when evaluating the efficacy of a classification model in classifying text. Precision is calculated using Eq. (15), and recall is specified in Eq. (16). Eq. (17) and (18) display the Accuracy and F-measure, respectively.

$$Precision = \frac{t_p}{t_p + f_p} \tag{15}$$

In the Eq. (15) t_p denotes the true positive rate and f_p shows the false positive rate in precision.

$$Recall = \frac{t_p}{t_p + f_n} \tag{16}$$

where t_p describes the true positive rate and f_n denotes the false negative rate in the recall.

Accuracy is defined as the ratio of the number of correctly classified objects to the total number of objects. Inaccuracy and true positive (t_p), true negative (t_n), false negative (f_n) and false positive (f_p) values are calculated as in Eq. (17):

$$Accuracy = \frac{t_p + t_n}{t_p + f_p + t_n + f_n} \tag{17}$$

The F-measure is the harmonic mean in which, precision and recall are combined, and the traditional f-measure is calculated as in Eq. (18):

$$F - measure = 2 \times \frac{p \times r}{p + r} \tag{18}$$

where p denotes the precision and r is the recall in the F-measure.

IV. Results

Different experiments were conducted to analyse the data, and the performance of RSS-LSTM was compared with different metaheuristic algorithms associated with LSTM such as GA-LSTM, PSO-LSTM, and FF-LSTM., for three datasets: 20 Newsgroups, Reuters-21578, and AG News were used to test the performance using several measurement parameters, including accuracy, F-measure, precision, and recall. The proposed model was tested using different sets of classes with different iterations.

A. Reuters-21578 Dataset Results

As shown in Table II, it is demonstrating that the proposed technique using RSS-LSTM performs more effectively than the other techniques. For Reuters-21578, RSS-LSTM's accuracy is superior to that of GA-LSTM, PSO-LSTM, and FF-LSTM. Compared to earlier methods, RSS-LSTM significantly outperformed the other methods on the entire dataset. On the Reuters-21578 text dataset, RSS-LSTM produced an accuracy of 96%, GA-LSTM produced 78%, firefly produced 56%, and PSO-LSTM produced 87% as shown in Fig. 3. Compared to the F-measure, RSS-LSTM outperformed GA-LSTM, PSO-LSTM and FF-LSTM, and achieved 95%, 64%, 86% and 54% respectively as shown in Fig. 4. Fig. 5 shows that RSS-LSTM delivered improved results in terms of precision compared to the existing techniques, RSS-LSTM achieved 96%, whereas GA-LSTM, PSO-LSTM and RR-LSTM achieved 72%, 86% and 71% respectively. In the recall scenario, Fig. 6 shows that RSS-LSTM again performs better than GA-LSTM, PSO-LSTM, and FF-LSTM and achieved results as of 96%, 68%, 87% and 56% respectively. The accuracy of the proposed method was superior to that of the other mentioned techniques, as demonstrated in Table II, for the entire dataset. When compared to the GA-LSTM, PSO-LSTM, and FF-LSTM algorithms, the RSS-LSTM approach performed better in terms of accuracy, precision, recall, and f-measure.

TABLE II. PERFORMANCE OF RSS-LSTM AMONG PSO-LSTM AND GA-LSTM USING REUTERS-21578 DATASET

Classifier	Measure criteria			
	Accuracy	F-measure	Precision	Recall
GA-LSTM	0.78	0.64	0.72	0.68
PSO-LSTM	0.87	0.86	0.86	0.87
FF-LSTM	0.56	0.54	0.71	0.56
RSS-LSTM	**0.96**	**0.95**	**0.96**	**0.96**

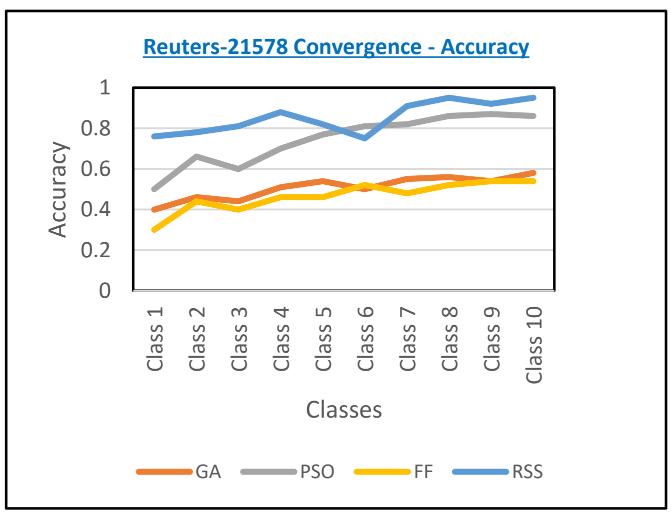

Fig. 3. Reuters-21578 convergence - accuracy.

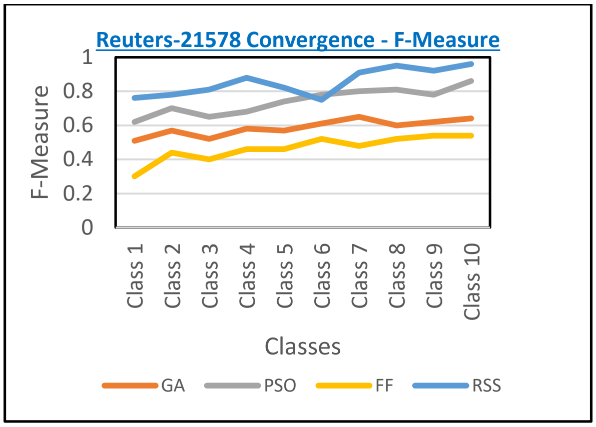

Fig. 4. Reuters-21578 convergence - f-measure.

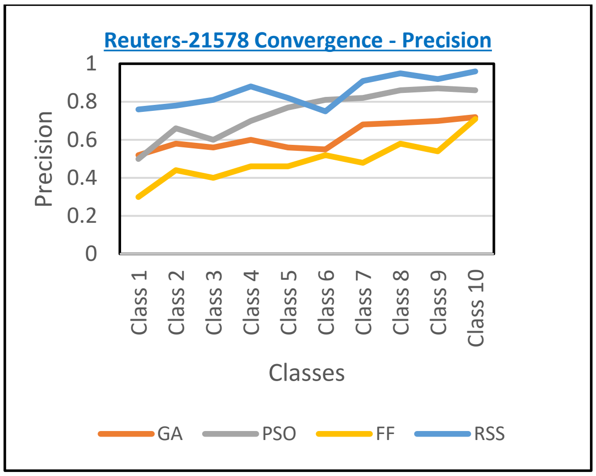

Fig. 5. Reuters-21578 - precision.

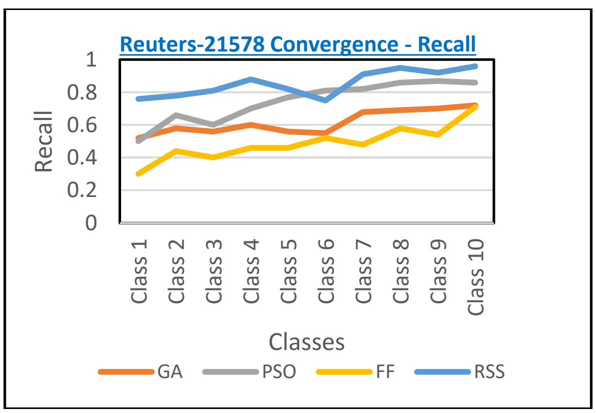

Fig. 6. Reuters-21578 convergence - recall.

B. Result of 20 Newsgroups Dataset

An experiment using 20 Newsgroups text datasets demonstrated that the proposed RSS-LSTM technique outperformed the GA-LSTM, PSO-LSTM, and FF-LSTM strategies already in use. Fig. 7 shows that the RSS-LSTM generated high accuracy of 77%. While GA-LSTM produced 49%, PSO-LSTM and FF-LSTM provided results of 58% and 21%, respectively as shown in Fig. 7 and Table III. In addition, the F-measure was 77% for RSS-LSTM, compared to 41%, 56%, and 35% for GA-LSTM, PSO-LSTM, and FF-LSTM, respectively as shown in Fig. 8. The RSS-LSTM technique provided better results as of 78% than the existing techniques for 20 Newsgroups dataset in precision comparison as shown in Fig. 9, while GA-LSTM, PSO-LSTM, and FF-LSTM achieved precisions as 50%, 73%, and 21% respectively. Similarly, Fig. 10 shows that RSS-LSTM achieved a higher recall as of 77% in comparison to GA-LSTM, PSO-LSTM and FF-LSTM. RSS-LSTM generated the best accuracy of 77%. The outcome for RSS-LSTM is superior to that of the GA-LSTM, PSO-LSTM, and FF-LSTM methodologies, as shown in Table III and described below.

TABLE III. PERFORMANCES OF RSS-LSTM RESULTS AMONG GA-LSTM, PSO-LSTM, AND FF-LSTM USING THE 20-NEWSGROUP DATASET

Classifier	Measure criteria			
	Accuracy	**F-measure**	**Precision**	**Recall**
GA-LSTM	0.49	0.41	0.50	0.49
PSO-LSTM	0.58	0.56	0.73	0.58
FF-LSTM	0.21	0.35	0.21	0.30
RSS-LSTM	**0.77**	**0.77**	**0.78**	**0.77**

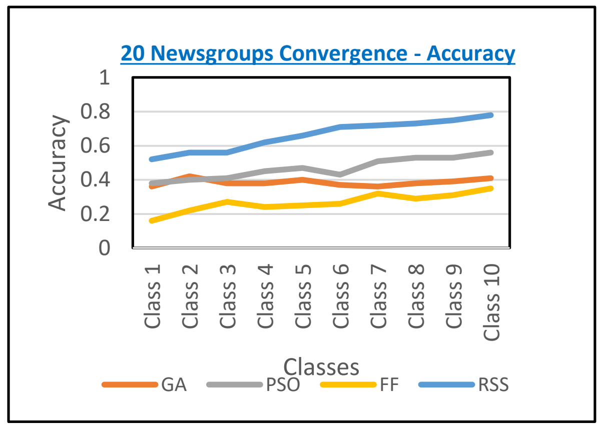

Fig. 7. 20 Newsgroups convergence – accuracy.

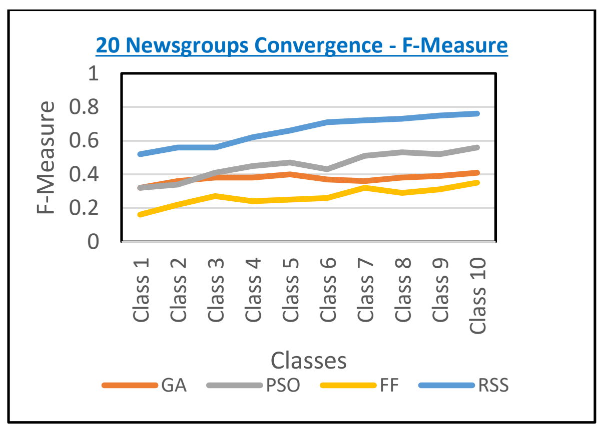

Fig. 8. 20 Newsgroups convergence - f-measure.

C. Result of AG News Dataset

Table IV shows the performance measured using the AG News dataset. The performance of the RSS-LSTM optimization approach was tested against those of existing GA-LSTM, PSO-LSTM, and FF-LSTM techniques. The study was carried out for the evaluation matrix, as accuracy, F-measure, precision, and recall are among the metrics used to evaluate RSS-LSTM. Compared to existing GA-LSTM, PSO-LSTM, and FF-LSTM techniques, the RSS-LSTM technique produced greater accuracy than other comparing techniques. Fig. 11 shows that GA-LSTM achieved 86% accuracy, PSO-LSTM and FF-LSTM produced 88% and 80% accuracy, respectively, while the proposed RSS-LSTM produced 91% accuracy.

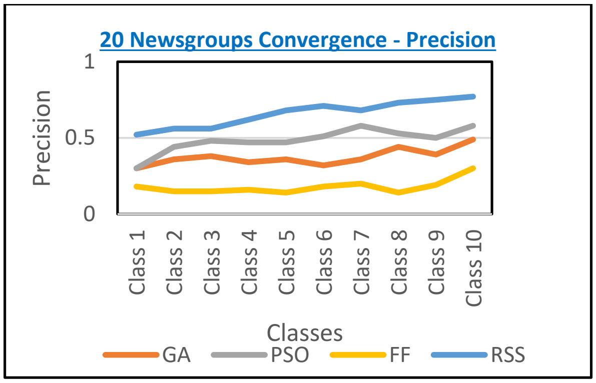

Fig. 9. 20 Newsgroups convergence – precision.

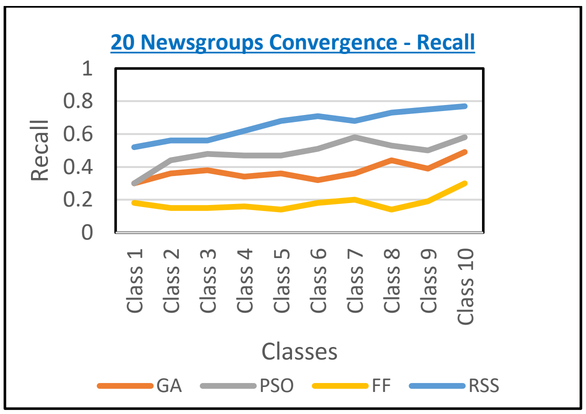

Fig. 10. 20 Newsgroups convergence – recall.

TABLE IV. PERFORMANCES RSS-LSTM RESULT AMONG GA-LSTM, PSO-LSTM, AND FIREFLY-LSTM USING THE AG NEWS DATASET

Classifier	Measuring Criteria			
	Accuracy	**F-measure**	**Precision**	**Recall**
GA-LSTM	0.86	0.85	0.86	0.86
PSO-LSTM	0.88	0.84	0.89	0.89
FF-LSTM	0.80	0.78	0.88	0.88
RSS-LSTM	**0.91**	**0.90**	**0.94**	**0.91**

To evaluate the F-measure score, significant results were obtained for RSS-LSTM, GA-LSTM, PSO-LSTM, and FF-LSTM, which were 91 %, 85%, 84%, and for FF-LSTM, 78% shown in Fig. 12. Additionally, RSS-LSTM achieved higher precision and Fig. 13 demonstrated 94% precision compared to 86%, 89%, and 88% precision for GA-LSTM, PSO-LSTM, and FF-LSTM, respectively. The outcome of RSS-LSTM was also measured for recall, and it provided a result of 91%, compared to 86%, 89%, and 88% for GA-LSTM, PSO-LSTM, and FF-LSTM, respectively also shown in Fig. 14. The table below shows that the overall performance of RSS-LSTM is superior to that of the GA-LSTM, PSO-LSTM, and FF-LSTM techniques.

Table V presents the combined results of the proposed RSS-LSTM model compared with GA-LSTM, PSO-LSTM, and GA-LSTM. Additionally, Fig. 15, 16, and 17 provide graphical representations of the results achieved by the proposed model and the compared techniques.

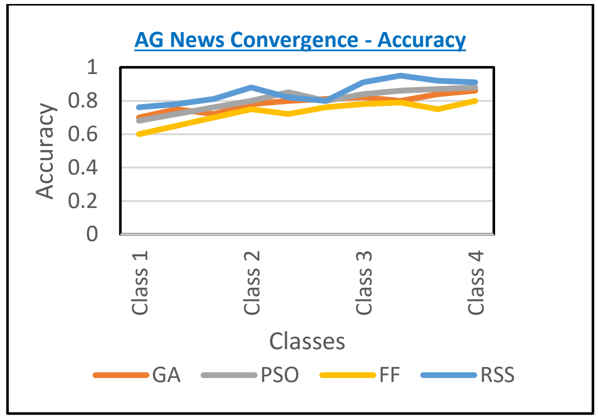

Fig. 11. AG News convergence – accuracy.

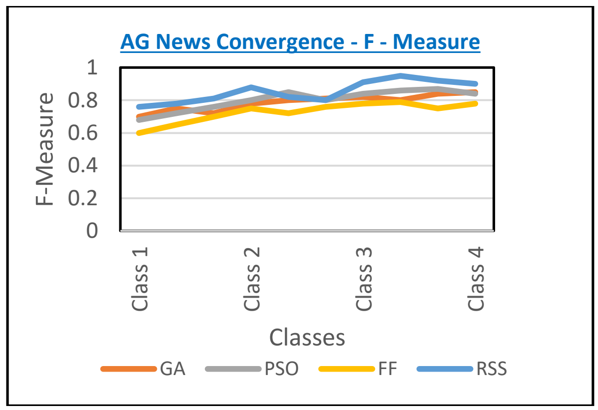

Fig. 12. AG News convergence - f – measure.

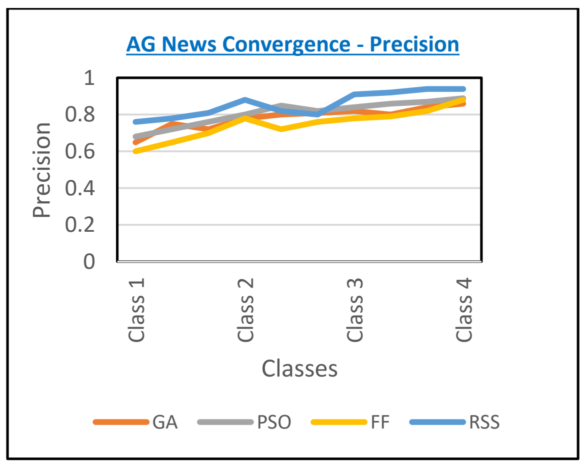

Fig. 13. AG News convergence - precision.

TABLE V. PERFORMANCES RSS-LSTM WITH RESPECT TO THREE DATASETS

Dataset	Technique	Measuring Matrix			
		Accuracy	F-Measure	Precision	Recall
Reuters-21578	GA-LSTM	0.78	0.64	0.72	0.68
	PSO-LSTM	0.87	0.86	0.86	0.87
	FF-LSTM	0.56	0.54	0.71	0.56
	RSS-LSTM	**0.96**	**0.95**	**0.96**	**0.96**
20 Newsgroups	GA-LSTM	0.49	0.41	0.50	0.49
	PSO-LSTM	0.58	0.56	0.73	0.58
	FF-LSTM	0.21	0.35	0.21	0.30
	RSS-LSTM	**0.77**	**0.77**	**0.78**	**0.77**
AG News	GA-LSTM	0.86	0.85	0.86	0.86
	PSO-LSTM	0.88	0.84	0.89	0.89
	FF-LSTM	0.80	0.78	0.88	0.88
	RSS-LSTM	**0.91**	**0.90**	**0.94**	**0.91**

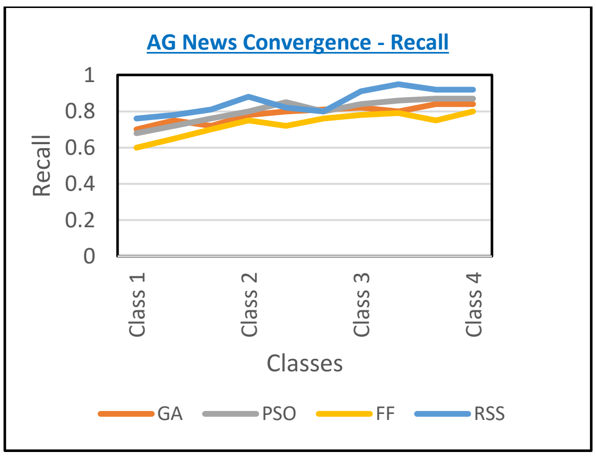

Fig. 14. AG News convergence – recall.

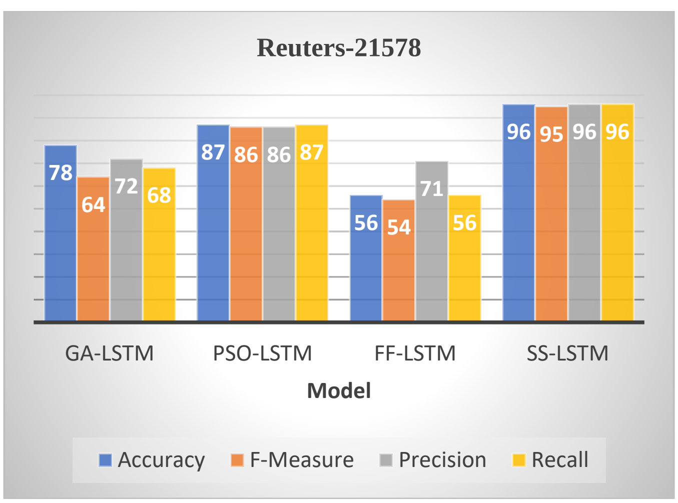

Fig. 15. Comparison with reuters-21578.

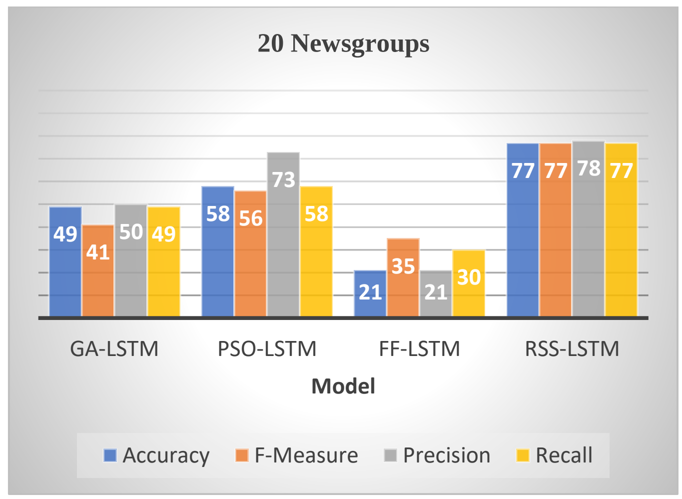

Fig. 16. Comparison with 20 newsgroups.

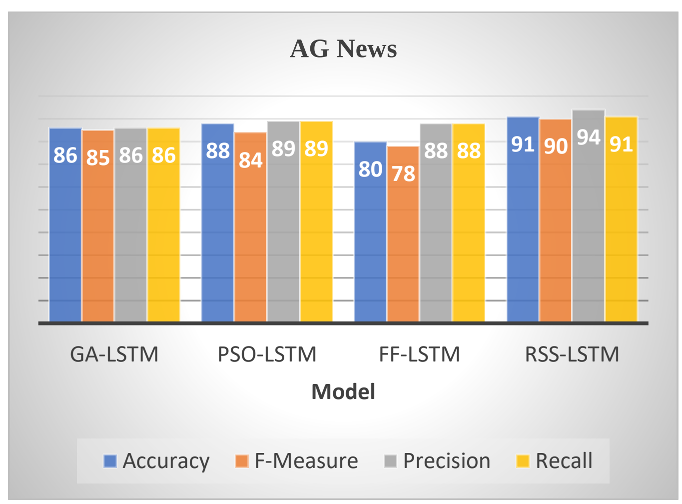

Fig. 17. Comparison with AG news.

D. Comparison of Models

The proposed RSS-LSTM model was also compared with state-of-the-art deep learning models, such as Support Vector Machine, Stochastics Gradient Descent (SGD), Random Forest (RF), Logistics Regression (LR), K-nearest Neighbour (KNN), Naïve Base (NB), Decision Tree (DT), Autor encoder (AE), AdaBoost (AB) using Reuters-21578, 20 Newsgroups and AG news dataset, where Proposed approached shows the significant results compare to mentioned techniques. Table VI describes the comparison of proposed technique with state of art models used for hyperparameter optimization using different datasets.

Above mentioned Table VII lists the hyperparameters with their ranges that we use in this study.

TABLE VI. PERFORMANCES RSS-LSTM ACCORDING WITH RESPECT TO THREE DATASETS

Ref	Year	Technique	Dataset	Findings
[63]	2023	Support Vector Machine, Stochastic Gradient Descent (SGD), Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbor (KNN)	Reuters-21578	Accuracy achieves using Reuters-21578 dataset as 0.8516, 0.8476, 0.8470, 0.8110, 0.8183, 0.8135
[64]	2022	OPT 175b, Bloom 176B, OPT 30b, OPT 1.3b	AG News	AG news achieve the accuracy as follows using different techniques 68.7, 39.5, 60.7, 37.6
[65]	2020	Naïve Base, SVM, Gradient, Boosting, Random Forest, Logistics Regression	20NewsGroups	Achieve maximum accuracy 67.3, 65.3,59.5,60.1 and 67.4 respectively
[66]	2020	Logistic Regression (LR), Decision Trees (DT), Support Vector Machine, AdaBoost (AB), Random Forest (RF), Multinomial Naïve Bayes (MNB), Multilayer Perceptron (MLP), Gradient Boosting (GB).	20NewsGroups	68.28, 44.44, 70.03, 45.61, 62.28 60.62, 60.12, 69.46
[67]	2017	Autoencoder	20NewsGroups	Achieve accuracy 73.78
Proposed Technique		**RSS-LSTM**	**Reuters, 20Newsgroups, AG News**	**Achieve maximum accuracy of 96%, 77%, and 91% respectively**

TABLE VII. HYPERPARAMETER RANGES

Model	Hyperparameter	Optimization Range
GA-LSTM	Dense units	16 to128
	Learning rate	0.001 to 0.1
	Dropout rate	0.1 to 0.05
PSO-LSTM	Dense units	16 to 128
	Learning rate	0.001 to 0.1
	Dropout rate	0.1 to 0.05
FF-LSTM	Dense units	16 to 128
	Learning rate	0.001 to 0.1
	Dropout rate	0.1 to 0.05
	Dense units	16 to 128
RSS-LSTM	Dense units	16 to 128
	Learning rate	0.001 to 0.1
	Dropout rate	0.1 to 0.05
	Dense units	16 to 128

TABLE VIII. TIME CONSTRAINTS OF HYPERPARAMETER OPTIMIZATION

Algorithm	Complexity	Dataset	Time (std dev)
PSO	O(I×(P×L+P))	Reut-21578	132.49
		20NG	71.23
		AG News	448.02
GA	O(G×P×L)	Reut-21578	220.28
		20NG	119.23
		AG News	500.64
FFA	O(num iterations ×num_fireflies^2 × len(firefly-bounds))	Reut-21578	279.25
		20NG	248.31
		AG News	2253.90
RSS	O(max-iterations × num-lairs × (len(search-space) + max-len))	Reut-21578	110.05
		20NG	67.23
		AG News	836.20

Table VIII describes the complexity equivalences, for PSO I, P, and L indicating the total number of iterations, number of particles in a swarm, and search space (dimensioned of parameters) respectively. For the GA algorithm, G indicates the generation (iteration), P resents the Population size and L resents the number of genes in an individual. The Firefly Algorithm contains a total number of iterations, the number of flies, and the length of parameters for a search space. Ringed Seal Search (RSS) consists of its number of iterations, the number of lairs that describe the search areas, the search space that defines the number of dimensions, and at last, it adds a maximum number of lairs. Additionally, Table VIII shows the standard deviation of temporal demands for each algorithm according to the dataset, low standard deviation values indicate more consistent performance, and they shed light on how variable or consistent the algorithm's execution times are across various datasets, however, these values may also vary depending at the factors such as algorithm’s number of iterations, number of epochs, batch size and other parameters.

Table IX demonstrates the variation of hyperparameters for different datasets.

TABLE IX. VARIATION OF HYPERPARAMETERS USING DIFFERENT DATASETS

Dataset	Technique	Dense Unit	Dropout	Learning Rate
Reuters-21578	GA-LSTM	118	0.3914	0.0087
	PSO-LSTM	16	0.2271	0.0364
	FF-LSTM	37	0.2221	0.0031
	RSS-LSTM	27	0.3654	0.01
20 Newsgroups	GA-LSTM	89	0.1527	0.0942
	PSO-LSTM	120	0.3515	0.0090
	FF-LSTM	34	0.2316	0.0093
	RSS-LSTM	54	0.2681	0.01
AG News	GA-LSTM	61	0.1635	0.0150
	PSO-LSTM	91	0.3527	0.0014
	FF-LSTM	104	0.3998	0.0087
	RSS-LSTM	122	0.15739	0.0069

V. DISCUSSION AND CONCLUSION

In diverse fields such as bioinformatics, sentiment analysis, online handwritten recognition, and text classification, LSTM is used to apply diverse classification issues. One area where academics are attempting to increase classification accuracy is text classification. Different experiments were conducted to analyse the data, and the performance of RSS-LSTM was compared with that of GA-LSTM, PSO-LSTM, and FF-LSTM. Three datasets, including 20 Newsgroups, Reuters-21578, and AG News, were used to test the performance using several measurement parameters, including accuracy, F-measure, precision, and recall. The proposed model was tested using different sets of classes.

The results presented in Fig. 3, 4, 5, and 6 demonstrate that the proposed approach RSS-LSTM outperforms existing methods, achieving 96% accuracy, 96% F-score, 95% precision, and 96% recall on the Reuters-21578 dataset. Similarly, as shown in Fig. 7, 8, 9, and 10, the proposed method RSS-LSTM outperforms existing approaches on the 20 News dataset, achieving 77% accuracy, 77% F-score, 78% precision, and 77% recall. Furthermore, Fig. 11, 12, 13, and 14 indicate that the proposed approach surpasses existing methods on the AG News dataset, with 91% accuracy, 90% F-score, 94% precision, and 91% recall.

According to the literature review, search methods affect LSTM performance when solving text classification optimization problems. Therefore, to improve the LSTM parameters for enhanced text classification accuracy, this research presented an enhanced technique called RSS-LSTM, conducted using datasets Reuter-21578, 20 Newsgroups, and AG News Dataset, which was used to evaluate the effectiveness of the proposed model. The simulation results demonstrated that in terms of Accuracy, F-measure, Precision, and Recall, the proposed RSS-LSTM surpasses existing techniques. The experimental results on different classes of these three datasets showed that the proposed model performed well in terms of term precision, F-value, precision, and recall. The proposed model also compares with LSTM addresses different types of text classification problems in various fields such as bioinformatics, opinion mining, handwriting, and online recognition. One of the areas where scholastics are endeavouring to increase characterization accuracy is text classification.

A. Future Work

To evaluate the performance of the proposed model with different hyperparameter and ranges. To evaluate the proposed models at images datasets. To assess the effect of different iterations of different algorithms. The proposed technique performs very well as per the given measuring matrix, however detailed temporal demands may also be required in future work using different iteration and parameter settings evaluate the proposed technique with other deep learning models such as Recurrent Neural Network (RNN), Feedforward Neural Network (FNN), Gated Recurrent Unit (GRU) Autoencoders (AE).

REFERENCES

[1] Hassan, S.U., J. Ahamed, and K. Ahmad, Analytics of machine learning-based algorithms for text classification. Sustainable Operations and Computers, 2022. 3: p. 238-248.

[2] Roul, R.K., S.R. Asthana, and G. Kumar, Study on suitability and importance of multilayer extreme learning machine for classification of text data. Soft Computing, 2017. 21(15): p. 4239-4256.

[3] Janani, R. and S. Vijayarani, Automatic text classification using machine learning and optimization algorithms. 2021. 25(2 %J Soft Comput.): p. 1129–1145.

[4] Kowsari, K., et al., Text Classification Algorithms: A Survey. 2019. 10(4): p. 150.

[5] Wanjale, K., et al. Analyzing Machine Learning Algorithm for Breast Cancer Diagnosis. in International Conference on Computer Vision and Robotics. 2023. Springer.

[6] Liu, C. and X. Wang, Quality-related English text classification based on recurrent neural network. Journal of Visual Communication and Image Representation, 2020. 71: p. 102724.

[7] Minaee, S., et al., Deep learning--based text classification: a comprehensive review. 2021. 54(3): p. 1-40.

[8] Joshi, R., P. Goel, and R. Joshi. Deep learning for hindi text classification: A comparison. in Intelligent Human Computer Interaction: 11th International Conference, IHCI 2019, Allahabad, India, December 12–14, 2019, Proceedings 11. 2020. Springer.

[9] Huang, W., et al., LSTM with compensation method for text classification. 2021. 20(2): p. 159-167.

[10] Cai, J., et al. Deeplearning model used in text classification. in 2018 15th international computer conference on wavelet active media technology and information processing (ICCWAMTIP). 2018. IEEE.

[11] Wistuba, M., N. Schilling, and L. Schmidt-Thieme. Hyperparameter Optimization Machines. in 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). 2016.

[12] Shankar, K., et al., Hyperparameter Tuning Deep Learning for Diabetic Retinopathy Fundus Image Classification. IEEE Access, 2020. 8: p. 118164-118173.

[13] Claesen, M. and B.J.a.p.a. De Moor, Hyperparameter search in machine learning. 2015.

[14] Meraihi, Y., et al., Grasshopper Optimization Algorithm: Theory, Variants, and Applications. IEEE Access, 2021. 9: p. 50001-50024.

[15] Fatyanosa, T.N. and M. Aritsugi. Effects of the Number of Hyperparameters on the Performance of GA-CNN. in 2020 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT). 2020. IEEE.

[16] Sun, J., et al. Balancing exploration and exploitation in multiobjective evolutionary optimization. in Proceedings of the genetic and evolutionary computation conference companion. 2018.

[17] Yang, X.-S., S. Deb, and S.J.a.p.a. Fong, Bat algorithm is better than intermittent search strategy. 2014.

[18] Qian, L., et al., A new method of inland water ship trajectory prediction based on long short-term memory network optimized by genetic algorithm. 2022. 12(8): p. 4073.

[19] Hall, M., et al., The WEKA data mining software: an update. ACM SIGKDD explorations newsletter, 2009. 11(1): p. 10-18.

[20] Sharif, W., et al., An Optimised Support Vector Machine with Ringed Seal Search Algorithm for Efficient Text Classification. Journal of Engineering Science and Technology, 2019. 14: p. 1601-1613.

[21] Wang, Z., et al. Automatic hyperparameter tuning of machine learning models under time constraints. in 2018 IEEE international conference on big data (Big Data). 2018. IEEE.

[22] Elgeldawi, E., et al. Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis. in Informatics. 2021. MDPI.

[23] Tellez, E.S., et al., An automated text categorization framework based on hyperparameter optimization. Knowledge-Based Systems, 2018. 149: p. 110-123.

[24] Isa, S.M., et al., Optimizing the hyperparameter of feature extraction and machine learning classification algorithms. 2019. 10(3): p. 69-76.

[25] Wu, H., et al., Review of Text Classification Methods on Deep Learning. 2020. 63(3).

[26] Duan, H.K., et al., Enhancing the government accounting information systems using social media information: An application of text mining and machine learning. 2023. 48: p. 100600.

[27] Zhou, H. Research of text classification based on TF-IDF and CNN-LSTM. in Journal of Physics: Conference Series. 2022. IOP Publishing.

[28] Zhai, Z., et al., Text classification of Chinese news based on multi-scale CNN and LSTM hybrid model. 2023: p. 1-14.

[29] Attieh, J. and J.J.K.-B.S. Tekli, Supervised term-category feature weighting for improved text classification. 2023. 261: p. 110215.

[30] Liu, S., C. Gao, and Y.J.a.p.a. Li, Large Language Model Agent for Hyper-Parameter Optimization. 2024.

[31] Kaur, S., et al., Hyper-parameter optimization of deep learning model for prediction of Parkinson’s disease. 2020. 31: p. 1-15.

[32] Springenberg, J., et al., Striving for Simplicity: The All Convolutional Net. 2014.

[33] Janani, R. and S.J.S.C. Vijayarani, Automatic text classification using machine learning and optimization algorithms. 2021. 25: p. 1129-1145.

[34] Ahmad, S.R., A.A. Bakar, and M.R. Yaakub. Metaheuristic algorithms for feature selection in sentiment analysis. in 2015 Science and Information Conference (SAI). 2015. IEEE.

[35] Vyas, P., J. Liu, and O. El-Gayar, Fake news detection on the web: An LSTM-based approach. 2021.

[36] Ji, Y., A.W.-C. Liew, and L.J.I.A. Yang, A novel improved particle swarm optimization with long-short term memory hybrid model for stock indices forecast. 2021. 9: p. 23660-23671.

[37] Liu, G. and J.J.N. Guo, Bidirectional LSTM with attention mechanism and convolutional layer for text classification. 2019. 337: p. 325-338.

[38] Singh Sachan, D., M. Zaheer, and R.J.a.e.-p. Salakhutdinov, Revisiting LSTM Networks for Semi-Supervised Text Classification via Mixed Objective Function. 2020: p. arXiv: 2009.04007.

[39] Sendhilkumar, S.J.E.S.w.A., Developing a conceptual framework for short text categorization using hybrid CNN-LSTM based Caledonian crow optimization. 2023. 212: p. 118517.

[40] Nassef, A.M., et al., Review of Metaheuristic Optimization Algorithms for Power Systems Problems. 2023. 15(12): p. 9434.

[41] Qorib, M., et al., Covid-19 vaccine hesitancy: Text mining, sentiment analysis and machine learning on COVID-19 vaccination Twitter dataset. 2023. 212: p. 118715.

[42] Amini, A. and R. Kalantari, Gold price prediction by a CNN-Bi-LSTM model along with automatic parameter tuning. 2024. 19(3): p. e0298426.

[43] Maulani, A.A., et al., Comparison of Hyperparameter Optimization Techniques in Hybrid CNN-LSTM Model for Heart Disease Classification. Sinkron : jurnal dan penelitian teknik informatika, 2024. 8(1): p. 455-465.

[44] Lilhore, U.K., et al., Hybrid CNN-LSTM model with efficient hyperparameter tuning for prediction of Parkinson’s disease. Scientific Reports, 2023. 13(1): p. 14605.

[45] Abumohsen, M., A.Y. Owda, and M. Owda, Electrical Load Forecasting Using LSTM, GRU, and RNN Algorithms. Energies, 2023. 16(5): p. 2283.

[46] Ben, J., et al., Attention-LSTM architecture combined with Bayesian hyperparameter optimization for indoor temperature prediction. Building and Environment, 2022. 224: p. 109536.

[47] Bakhashwain, N. and A. Sagheer, Online Tuning of Hyperparameters in Deep LSTM for Time Series Applications. International Journal of Intelligent Engineering & Systems, 2021. 14(1).

[48] Li, Y., Y. Zhang, and Y. Cai, A New Hyper-Parameter Optimization Method for Power Load Forecast Based on Recurrent Neural Networks. Algorithms, 2021. 14(6): p. 163.

[49] Li, W., et al., HELP: An LSTM-based Approach to Hyperparameter Exploration in Neural Network Learning. Neurocomputing, 2021. 442.

[50] Kumar, P., S. Batra, and B. Raman, Deep neural network hyper-parameter tuning through twofold genetic approach. Soft Computing, 2021. 25(13): p. 8747-8771.

[51] Andonie, R. and A.-C. Florea, Weighted random search for CNN hyperparameter optimization. arXiv preprint arXiv:2003.13300, 2020.

[52] Aufa, B.Z., S. Suyanto, and A. Arifianto. Hyperparameter Setting of LSTM-based Language Model using Grey Wolf Optimizer. in 2020 International Conference on Data Science and Its Applications (ICoDSA). 2020.

[53] Gorgolis, N., et al. Hyperparameter optimization of LSTM network models through genetic algorithm. in 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA). 2019. IEEE.

[54] Nakisa, B., et al., Long short term memory hyperparameter optimization for a neural network based emotion recognition framework. IEEE Access, 2018. 6: p. 49325-49338.

[55] Loussaief, S. and A. Abdelkrim, Convolutional neural network hyper-parameters optimization based on genetic algorithms. International Journal of Advanced Computer Science and Applications, 2018. 9(10).

[56] Qian, L., et al., A New Method of Inland Water Ship Trajectory Prediction Based on Long Short-Term Memory Network Optimized by Genetic Algorithm. 2022. 12(8): p. 4073.

[57] Andersen, H., et al. Evolving neural networks for text classification using genetic algorithm-based approaches. in 2021 IEEE Congress on Evolutionary Computation (CEC). 2021. IEEE.

[58] Razno, M.J.C.L. and I. Systems, Machine learning text classification model with NLP approach. 2019. 2: p. 71-73.

[59] Mohapatra, P., et al., An improved cuckoo search based extreme learning machine for medical data classification. 2015. 24: p. 25-49.

[60] Chacra, D.A. and J. Zelek. Road Segmentation in Street View Images Using Texture Information. in 2016 13th Conference on Computer and Robot Vision (CRV). 2016.

[61] Aghaabbasi, M., et al., On hyperparameter optimization of machine learning methods using a Bayesian optimization algorithm to predict work travel mode choice. 2023. 11: p. 19762-19774.

[62] Saadi, Y., et al., Ringed seal search for global optimization via a sensitive search model. 2016. 11(1): p. e0144371.

[63] Daud, S., et al., Topic Classification of Online News Articles Using Optimized Machine Learning Models. 2023. 12(1): p. 16.

[64] Gonen, H., et al., Demystifying prompts in language models via perplexity estimation. 2022.

[65] Singh, A.K., <Text-classification-performance-analysis-using-dif-ferent-supervised-machine-learning-models.pdf>. International Journal of Scientific & Engineering ResearchISSN 2229-5518, September-2020. Volume 11(Issue 9).

[66] Alyasiri, O., et al., Wrapper and Hybrid Feature Selection Methods Using Metaheuristic Algorithms for English Text Classification: A Systematic Review. IEEE Access, 2022. 10: p. 39833-39852.

[67] Aziguli, W., et al., A Robust Text Classifier Based on Denoising Deep Neural Network in the Analysis of Big Data. Scientific Programming, 2017. 2017: p. 1-10.