SN Computer Science (2024) 5:67
https://doi.org/10.1007/s42979-023-02382-z

ORIGINAL RESEARCH

NDDSM: Novel Deep Decision-Support Model for Hate Speech Detection

Ashwini Kumar[1] · **Santosh Kumar**[1]

Received: 23 June 2023 / Accepted: 1 October 2023

Abstract

The pervasiveness of social media in people's lives is indisputable, the issue where it has become a necessary part of daily practices. However, unrestricted access to social media allows anonymous individuals to spread meaningless or even hostile information, defeating communication's purpose. Social media's positive and negative impact on society or individuals becomes more pronounced as its usage increases. As the harmful effect of unmonitored 'hate speech' becomes increasingly apparent, detecting such content has become a crucial concern in social media. In a recent study, machine-learning models have been developed to identify hate speech across multiple languages. As a result, the use of Bidirectional long short-term memory (Bi-LSTM) and convolutional neural network (CNN) for feature extraction in evaluating and identifying hate speech has risen. However, LSTM and CNN hyperparameters are typically selected based on expert opinion and prior research, making it difficult for the model to generalize since its creators need to know the optimal values for its parameters. To address this issue, we propose a novel deep decision support model which uses the sparrow search algorithm (SSA) to optimize the Bi-LSTM and CNN model hyperparameters for detecting hate speech. We employed the SSA for the decision support system to identify the best hyperparameters for the model architecture to improve its interpretability and accuracy. The benchmark datasets have been used to evaluate the model's performance, and the results indicate that our proposed model outperforms conventional hate speech detection systems.

Keywords Social media · Sparrow Search Algorithm · Hate speech · Decision support system

Introduction

The internet has demonstrated its effectiveness as a valuable instrument for facilitating public engagement. Social media platforms on the internet facilitate numerous significant dialogues. Furthermore, individuals depend on social media to communicate, effectively converting the world into a global community. However, the expeditious transition towards digitalization raises apprehensions, including the issue of hate speech. According to Mandl T. et al. [1], cyber aggression manifests aggressive or degrading conduct directed towards individuals on digital platforms. Hate speech is commonly motivated by race, nationality, religion, colour, gender, and other similar characteristics [2]. As per recent research findings [3], individuals are observed to generate nearly 3.3 million Facebook posts and 4.5 million tweets within a minute. Nevertheless, these figures are on the rise with each passing day. Through the process of analysis and extraction of reviews about specific topics such as influences, celebrities, or any issues, it becomes evident that a significant proportion of the vast corpus of tweets or Facebook posts often comprises expressions of hatred**.**

Various machine-learning techniques for detecting hate speech online have been suggested and widely researched. Deep learning has significantly advanced in several application areas in recent years. Numerous academic research works have demonstrated the capability of deep learning models to extract suitable features and generalize well. One such example is the LSTM approach, which effectively overcomes the issue of vanishing gradients by including memory units to capture long-term relationships [4]. This technique

This article is part of the topical collection "Advanced Computing and Data Sciences" guest edited by Mayank Singh, Vipin Tyagi and P.K. Gupta.

✉ Ashwini Kumar
ashwinikumar.cse@geu.ac.in

[1] Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun, India

Published online: 06 December 2023

beats memoryless classification approaches and standard recurrent neural networks (RNNs). As a result, LSTM is often utilised for time-series data processing, including textual data.

Despite the benefits of LSTM for solving time-series problems, it has some limitations. First, deep learning models such as LSTM must clearly explain the parameters used for prediction or the cause behind the final decision; and second, the model's hyperparameters are often set based on prior knowledge or expertise, which can introduce a degree of subjectivity. On the other hand, choosing appropriate hyperparameters can improve the neural network model's structural integrity, scalability, and overall performance. Thus, researchers are also interested in finding optimal network hyperparameters and reducing the impact of human factors.

Therefore, introducing optimal hyperparameters becomes essential for developing an efficient Decision Support System (DSS). To this, an SSA-BiLSTM-CNN has been proposed for hate speech detection. This study focuses on designing a hate speech detection model called SSA-Bi-LSTM-CNN. The Sparrow Search Algorithm optimizes the CNN and LSTM networks to address the abovementioned limitations. This model aims to identify the most significant parameter values for the CNN, LSTM, dense layer, and number of epochs using a unique optimization method based on population intelligence. This methodology also addresses the limitation of Deep Neural Networks (DNN) in explaining the underlying mechanism by which the model reaches conclusive decisions and is accessible to human-induced biases Furthermore, it presents a novel technique for optimizing the decision-making model in hate speech identification. The new approach utilizes the Sparrow Search Algorithm to maximize its efficient and optimal local search, ultimately leading to global optimization.

The contribution of this paper is described as follows:

Develop a novel SSA-BiLSTM-CNN model to improve the optimization of hyperparameters for investigating user-generated content and understanding complex textual features.
To improve the exploitation of the local search ability of the proposed model of a heuristic algorithm.
The SSA-BiLSTM-CNN is employed to enhance the model's accuracy by optimizing the optimal connection weights for the DNN model.

The paper is structured as follows: Related Work introduces related work on hate speech detection, while Experiments and Results discusses the multi-social media dataset used in this study. Next, the proposed SSA-BiLSTM-CNN model is presented in Results Analysis. Finally, Sect. 5 shows the results and performance of the model, and Conclusion describes the conclusion and future work.

Related Work

In this section, we review the relevant research and existing models for detecting hate speech discussed in Hate Speech Detection. We then introduce the SSA optimization algorithm in Sparrow Search Algorithm and compare the effectiveness of the SSA algorithm versus PSO in Comparative Study.

Hate Speech Detection

The emergence of the "social web" has dramatically accelerated advancements in NLP research. Through machine learning, deep learning, and natural language processing techniques, researchers have analysed the emotional context of posts on popular social media platforms such as Facebook and Twitter. This has turned the social web into a dynamic field of study. Platforms like Twitter, YouTube, and Facebook have become global forums where individuals from diverse linguistic, cultural, and socioeconomic backgrounds discuss various topics, including current events and popular culture. However, this increased diversity also poses a higher risk of encountering hate speech incidents [1, 2, 5, 6]. Managing and addressing such online behavior becomes challenging due to the complexities introduced by different languages and their nuances [7–9]. Artificial intelligence techniques, such as machine learning and deep learning, have gained significant attention for understanding the reasons behind labeling the text as hate speech in various contexts, including social media and medical domains.

Several studies have focused on the issue of online hate speech [5, 10–13]. This aggressive type of communication is typically seen as offensive and motivated by the author's prejudice against a specific group or individuals. The central aspect analyzed in this context is targeting, where the expression of hatred is directed towards a particular group or community, including refugees [14, 15]. Waseem et al. [16] have classified various types of abuse based on the target recipient, differentiating between those aimed at individuals/entities versus groups and the level of particularity used. Neural language models exhibit the potential to address this task; however, previous studies have utilized training data with a similarly expansive definition of hate speech. In addition, incorporating non-linguistic features such as the author's gender or ethnicity can enhance hate speech classification. However, obtaining or relying on this information from social media platforms is often achievable and unreliable. ElSherief et al. [17] investigated the connection between individuals who provoke hatred and their targets and their online prominence.

They examined hate speech on Twitter and identified the most used terms in hateful tweets using Hatebase, a repository of hate words. They introduced a significant hate dataset that does not have manual annotations, and all tweets containing specific hate words are considered hate speech. According to Salminen et al. [14], Different machine learning algorithms, such as XGBoost, SVM, LR, NB, and feed-forward neural networks, were employed and evaluated using various feature representations like Bag-of-Words (BoW), TF-IDF, Word2Vec, and BERT. Among these models, XGBoost presented the best performance, achieving an F1 score of 0.92 when evaluating all the features.

Analyzing the interactions among various groups is crucial to understand online hate speech. Considerable research studies have explored different aspects of hate speech, including the prevalence of hate groups and group discrimination [18, 19], the use of effective communication to promote hate speech [20], the influence of exposure to extremist content on social media leading to polarization [21], the spread of hate speech in society [22], and the effects of social exclusion [23, 24]. In addition, interpretive techniques are often utilized to capture the subtleties of hate speech, as contextual and subjective factors can have a significant impact. Wafa et al. [25] introduced a novel approach to hate speech detection that involves selecting features based on specific criteria to identify which characteristics are essential for the embedding strategy. Basak and colleagues created a web application called "block shame" to address various forms of abusive online behaviour, including comparison, sarcasm, whataboutery, and judgement. The software determines and blocks spammers and defines these behaviours. They trained a classifier to differentiate between varying degrees of cyberhate directed towards Black Minority Ethnic (BME) and religious communities on Twitter. The classifier successfully classified cyberhate into "moderate" and "extreme." A precision of 0.77 was attained in this classification task. Deep learning techniques involving recurrent neural networks have been suggested for smaller datasets. Utilizing a sophisticated attention mechanism along with multi-task learning has improved sentiment categorization by considering human sentiment. Sequeira et al. [26] employed various neural network models, including Long Short-Term Memory (LSTM) and Text Convolutional Neural Networks (TextCNN), along with language embedding methods to classify tweets related to drug misuse. The accuracy of the proposed model achieved 0.83 on the Twitter and Reddit datasets. But still, the proposed model needs to perform a significant result. Zhao et al. [27] introduce the Deep Neural Network AE-based Hate Speech Identification has effectively handled challenging data. They collected tweets by searching for keywords such as "bully," "bullied," and "bullying." Consequently, their dataset primarily consisted of reports or individual accounts of cyberbullying rather than actual instances of cyberbullying. Additionally, their technique implies that the training dataset would not include cyberbullying signals lacking these specific keywords. Wang et al. [28] introduced a novel approach called the local CNN-LSTM model with a tree structure for emotional analysis. Unlike conventional CNN models that treat the entire text as input, this model divides the text into multiple regions and extracts valuable emotional information from each region, assigning weights accordingly. Instead of using the entire text, the proposed restricted CNN focuses on specific portions of the text as regions. By combining CNN and LSTM, the model improves classification accuracy by considering long-distance dependencies. These hybrid CNN-LSTM models proposed in this study have achieved superior results compared to previous approaches.

A possible way to enhance the efficiency of hate speech identification is using LSTM and BiLSTM models, which allow for the examination of the sequential structure of the data. Ahmed et al. [20] conducted experiments on Bangla text to identify cyberbullying. They employed LSTM, BiLSTM, and GRU models for this purpose. Among the various models tested, Multinomial Naïve Bayes demonstrated the highest performance for datasets comprising Romanized Bangla texts. It achieved 84% accuracy, for these datasets. Dadvar et al. [21] explored the identification of cyberbullying in YouTube comments by evaluating four deep learning models. They utilized GloVe and random word embeddings in their experiments. The findings indicated that the BiLSTM model outperformed traditional machine learning algorithms in terms of performance. This section of the paper thoroughly analyses several scholarly works that categorize hate speech based on its textual content.

Sparrow Search Algorithm

In the sparrow search algorithm [29], the sparrows are divided into three categories, producers who look for potential food, scroungers who follow the producers to eat food, and monitors who look out for the hunters or enemies and alert the producers and scroungers to protect themselves.

Initially, the sparrow population is initialized randomly, and over each iteration, the algorithm evaluates the fitness of each sparrow by measuring the objective function value at its location. Then, it updates the position of each sparrow based on a combination of its previous position, the best position found so far, and the positions of other sparrows in its neighbourhood. The process of evaluating the fitness of each sparrow, updating their positions, and introducing random searches is repeated for a certain number of iterations or until a stopping criterion is met. Finally, the best solution found by the algorithm is returned as a result.

Assume that the total population of sparrows is N in a D-dimensional space. The D will represent the number of hyperparameters that we require to optimize. The position of each bird can be represented in $N * D$ dimensional matrix. For instance, let us suppose $X(i,j)$ represents position of i^{th} sparrow in j^{th} dimension, where i can be any integer between 0 and N, j can be any integer between 0 and D.

For each iteration, the value of each bird is updated based on its fitness value determined by fitness function. The sparrow with a high fitness level, meaning they are in the top 20% of the population, are selected as producers. Then, the producers are updated using the formula below.

$$X_{i,j}^{c+1} = \begin{cases} X_{i,j}^{c} * exp\left(\frac{-i}{\alpha.c_{max}}\right), & A_V S_T \\ X_{i,j}^{c} + r.L, & A_V \geq S_T \end{cases} \quad (1)$$

The variable "c_{max}" represents the current iteration number, while "α" is a uniform random number between 0 and 1. In this context, "S_T" denotes the upper limit of the number of iterations, and "r" is a random number with a standard normal distribution. "L" is a 1 × D matrix, where all the elements are set to 1."A_V" is an alarm value, which can take a value between 0 and 1, and "S_T" is the safety threshold, which can range from 0.5 to 1. If A_V is less than "S_T", it means that there is no predator, and a large-scale search can be conducted. However, if "Av" is greater than or equal to "S_T", it means that the predator has been detected, and the population should relocate to a secure location upon receiving the warning signal to ensure safety.

The scroungers update their location by following the producers and utilizing the following formula to obtain food.

$$X_{i,j}^{c+1} = \begin{cases} Q * exp\left(\frac{X_{worst}^{c} - X_{i,j}^{c}}{i^2}\right), & i < \frac{n}{2} \\ X_{p}^{c+1} + \left|X_{i,j}^{c} - X_{p}^{c+1}\right|.A^{+}.L, & i \leq \frac{n}{2} \end{cases} \quad (2)$$

The equation involves multiple variables, such as X_p, which denotes the best possible position for the producer, and X_{worst}, which indicates the most impoverished global status at the current time. A is a matrix with random elements of either 1 or -1, and it satisfies a specific equation. When the index i is greater than half the total number of scroungers $\frac{n}{2}$, it means that the i^{th} scrounger is hungry, has low energy, and has a poor fitness level, so it needs to move to a different location to find food. The scrounger will follow the producer in the best position to search for food. A^+ is a matrix determined by the equation $A^+ = A^T(AA^T)^{-1}$.

The equation using which monitors are updated using the following equation 3.

$$X_{i,j}^{c+1} = \begin{cases} X_{best}^{c} + \beta.\left|X_{i,j}^{c} - X_{best}^{c}\right|, & f_i > f_g \\ X_{i,j}^{c} + K.\left(\frac{\left|X_{i,j}^{c} - X_{worst}^{c}\right|}{(f_i - f_w) + \epsilon}\right), & f_i = f_g \end{cases} \quad (3)$$

In the given context, X_{best} indicates the most favorable position within the entire area. β and K represent control parameters that regulate the step size and are randomly generated from a standard normal distribution and a range between -1 and 1, respectively. The fitness value of the sparrow at present is indicated as f_i, while f_g and f_w signify the best and worst fitness values of positions in the whole area, respectively. To prevent division by zero, an extremely small constant ε is used. If the value of f_i is greater than f_g, it implies that the sparrow is situated at the edge of the population and is vulnerable to predators. In such a situation, the sparrow needs to approach other members of the group to ensure safety.

Comparative Study

This section discusses the effectiveness of the Sparrow search algorithm (SSA) vs Particle Swarm Optimization (PSO) in solving optimization problems. The Sparrow search algorithm is relatively new, and much research still needs to be done comparing its performance to other algorithms, including PSO [30]. However, the Sparrow algorithm has shown promising results in some recent studies [31–33]. This strategy enhanced the model's predictive accuracy and has been effectively utilized in stock price prediction tasks, yielding favorable outcomes. Rajathi et al. [32] introduced a novel model using a sparrow search algorithm for diagnosing brain tumors using the Internet of Medical Things (IoMT) and cloud technology. This IoMTC-HDBT model incorporates a functioning link neural network (FLNN) capable of detecting and categorising MRI brain images as normal or abnormal, stimulating early diagnosis and enhancing healthcare quality. The model's validation is performed using the BRATS2015 Challenge dataset, and the experimental investigation evaluates its sensitivity, accuracy, and specificity. The experimental results indicate the outstanding performance of the proposed model, performing an accuracy rate of 0.984. Zhang et al. [33] introduce a novel CSSA-SCN procedure, which converges the chaotic sparrow search algorithm with the stochastic configuration network (SCN). This procedure aims to enhance the regression performance of SCN, mainly when dealing with large-scale data problems. Techniques such as logistic mapping, self-adaptive

hyper-parameters, and a mutation operator are integrated into the sparrow search algorithm to improve the optimization capabilities. By employing the chaotic sparrow search algorithm to optimize the selection of random parameters in SCN, the proposed CSSA-SCN model achieves an outstanding regression accuracy of 0.9. In addition, its unique features distinguish it from PSO, including combining a "smart mutation" operator that helps present diversity into the population and control premature convergence.

The particle swarm optimization (PSO) algorithm has been widely used and reviewed for several years, and it has shown acceptable performance in solving many optimization problems. However, like any optimization algorithm, PSO has restrictions and imperfections that should be evaluated when applying it to different issues.. Some of the rules of PSO include:

i. *Premature Convergence*: PSO can converge prematurely to a suboptimal solution because the swarm gets trapped in a local optimum. This PSO can be mitigated using a larger swarm size or incorporating adaptive strategies.
ii. *Limited diversity*: PSO convergence can lead to a lack of diversity in the population of solutions and may result in the algorithm missing better solutions further away from the swarm's current location.
iii. *Sensitivity to parameters*: PSO requires several parameters to be set, such as the swarm size, the learning factors, and the internal weight. The optimal parameter values may differ for different optimization problems, and finding these values can be time-consuming.
iv. *Not suitable for discrete optimization problems*: PSO is designed for continuous optimization problems and may perform poorly for discrete optimization problems where only specific values are allowed.
v. *No guarantee of global optimality*: Like many other optimization algorithms, PSO does not guarantee to find the global optimum and may get stuck in a local optimum.
vi. Limited scalability: PSO may need to scale better to high-dimensional problems due to the curse of dimensionality, which can result in slow convergence and high computational costs.

The sparrow search algorithm is newer than PSO, and it has been shown to outperform PSO in some cases, especially for problems with high-dimensional search spaces or multiple objectives. Additionally, the algorithm has a more diverse population and uses a more complex update rule that considers the different behaviour of sparrows during foraging. As a result, this SSA has shown promising results in optimizing classification models, particularly for binary classification problems. In addition, the algorithm's ability to adaptively adjust the search range and weight coefficients can lead to better optimization performance and faster convergence [31].

Proposed Novel Deep Decision Support Model (NDDSM)

This section presents enhancements to the standard Sparrow Search Algorithm (SSA). The SSA algorithm has a precise structure, simple execution, few control factors, and robust local searchability. The random initialization technique is used to determine the sparrow's original location. Although the initial positions are guaranteed to be random utilizing this method, the convergence speed and accuracy of the solution are decreased because some people's optimum initial locations deviate too much from the actual optimal positions. The SSA provides several optimization options, including fast convergence and high search accuracy. Introducing the Iteratively Randomized Local Search (IRLS) method in Algorithm 1 improves the exploitation phase of SSA to prevent getting trapped in local optima. Algorithm 2 outlines the optimized version of the SSA-BiLSTM-CNN algorithm, which begins by generating a swarm of N sparrows in D dimensions. The main loop then iterates through these N sparrows, adjusting their positions using Eqs. (1), (2), or (3). Eventually, Algorithm 2 returns the best optimal solution consisting of x^{best} and f_g.

Several recent studies [31–33] have successfully utilized an SSA in several branches of engineering, providing helpful context for our investigation. We focused on optimizing the network model's hyperparameters to reduce the impact of humans on the model and boost its capacity to make predictions, as opposed to the approach [17]. Therefore, we choose to optimize for the learning rate, the number of LSTM and CNN neurons, dense layer neurons, and the epoch number. To accomplish this alignment between data characteristics and model structure, the SSA was used to optimize these target parameters of LSTM. The interpretability of a deep neural network is improved by SSA's ability to discover the best values of network parameters quickly and effectively [31–33]. Furthermore, the most effective Deep Neural Network (DNN) weights correlate directly with the model's predicted outcomes. This research suggests a technique for SSA-based BiLSTM-CNN improvement.

The proposed SSA is combined with BiLSTM-CNN to determine the optimal DNN weights, improving the model's effectiveness. The enhanced model incorporates the best strategies used by sparrows to update positions, ensuring the most efficient way to achieve the optimal solution with minimum repetitions. The initial weight values are randomly generated within a specified range. The proposed

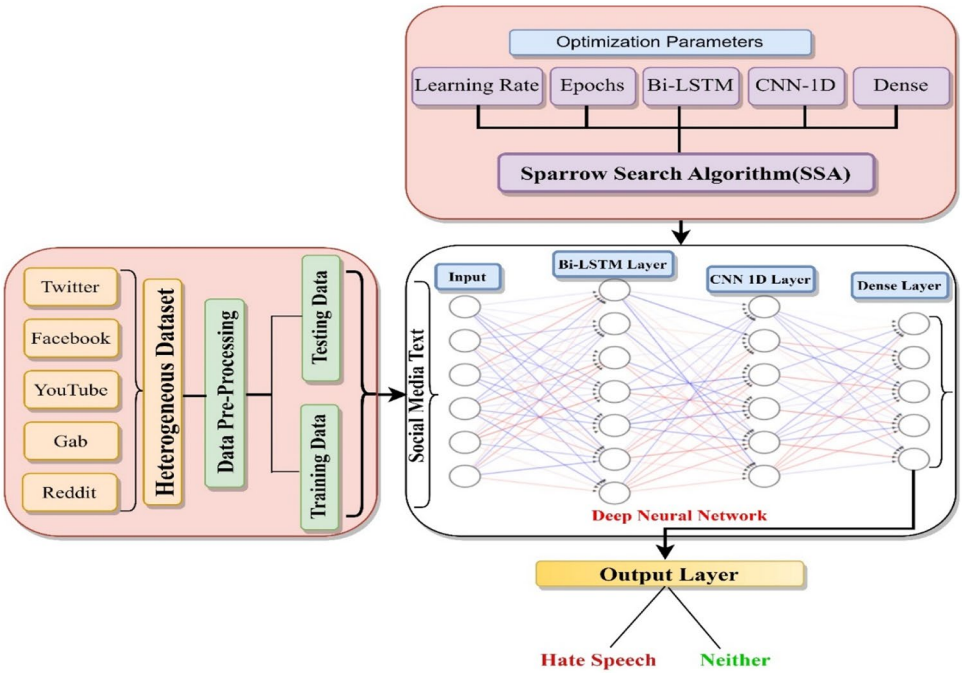

Fig. 1 Proposed NDDSM architecture

SSA-BiLSTM-CNN model for training our optimized Deep Neural Networks is shown in Fig. 1.

The network architecture denoted as SSA-BiLSTM-CNN comprises four distinct layers: the input layer, LSTM layer, hidden layer, and output layer. The SSA performs the optimization of the target parameters, namely the learning rate, epoch number, and neuron counts in the two hidden layers. The parameters' value range is specified, and the relevant parameters and the population's positional details are initialized randomly.

While exploring the field of sparrow searches, the sparrows may exceed the boundaries of the search space (*lower_bound, upper_bound*). This means that a sparrow might go beyond the range of the search space. However, repositioning individual sparrows to a random area within the search space can yield better results, given the stochastic nature of

```
Check_limits (new_pos, lb, ub)
// new_pos- current position of x_i^c, lb – lower_bound, ub - upper bound
if new_pos < lb then
    new_pos = lb+ (lb– new_pos)
else
    if new_pos > ub then
        new_pos = ub - (new_pos – ub)
    else
        return new_pos
    end if
    return new_pos
end if
```

Algorithm 1 IRLS Algorithm

Input: individuals N, dimension D, iterations c_{max}, producers PN, scroungers SN, monitor MN
Output: optimal value $\boldsymbol{x}^{best}$

Initialize individual optimal value f_i, global optimal value f_g
for c in c_{max} **do**
  for producer *i=1:PN* do
    Update producer at i^{th} location using Eq. (1)
    Apply *check_limits ()* on Producer *i*
    *// **use Algorithm 1***
  end for
  for scrounger *i=(PN+1): N* do
    Update scrounger at i^{th} location using Eq. (2)
    Apply *check_limits ()* on scrounger *i*
    *// **use Algorithm 1***
  end for
  for monitor *i=1: MN* do
    Update monitor at i^{th} location using Eq. (3)
    Apply *check_limits ()* on monitor *i*
    *// **use Algorithm 1***
  end for
  for *i =1: N* do
    Update f_i and f_g
  end for
end for
return the best optimal value $\boldsymbol{x}^{best}$

Algorithm 2 Optimization algorithm for SSA-BiLSTM-CNN

Table 1 Summary of existing heterogenous datasets from multi-social medias

Data Source	Paper	Year	ML Approach	Instance
Twitter	Davidson et al. [5]	2017	LR, SVM, DT, NB	24,783
Twitter	Thomas et al. [1]	2019	LSTM	7005
Twitter	Zampieri et al. [2]	2019	CNN	13,240
Twitter	Ousidhoum et al. [6]	2019	BiLSTM, BOW	5647
Twitter	Golbeck et al. [34]	2017	Corpus	20,360
Twitter	Fortuna et al. [35]	2018	Corpus	45,407
Facebook, YouTube	Chung et al. [36]	2019	Corpus	20,186
Facebook, YouTube	Salminen et al. [37]	2020	SVM, LR, DT, RF, Adaboost	3222
Gab	Kennedy et al. [38]	2022	Gab corpus	22,527
Reddit	Karrek et al. [39]	2020	Reddit Corpus	40,000
Total instances from multi-social medias				**202,377**

Table 2 Summary of Multi-Social Media Attributes

Class	Label	Total Records
Hate Speech	0	113,651
Neither	1	88,726

meta-heuristics. Thus, we have incorporated this technique as an upgrade step to address the misuse cycling of sparrows and correct the randomness that the original SSA algorithm might cause. Additionally, we have proposed a new method to enhance the optimum solution to prevent the SSA from being stuck in local optima during the exploitation phase. We have presented the pseudo-code for the Iteratively Randomized Local Search (IRLS) method in Algorithm 1.

The architecture and decision-making capabilities of the SSA-BiLSTM-CNN model are shaped by the parameters optimized using an algorithm. These optimized parameters are utilized in constructing the model. Our parameter choices and model creation process involve an iterative algorithmic optimization search that reduces the human influence and simplifies the interpretation of the model's structure and parameters. The proposed SSA-BiLSTM-CNN model is described using pseudocode in Algorithm 2.

Experiments and Results

This section includes comprehensive descriptions of the datasets, data pre-processing, optimization of the model hyperparameters, and analysis of the results.

Dataset Description

In this section, we utilized ten pre-existing datasets that had already been annotated [1, 2, 5, 6, 34–40]. Out of these, six were obtained from Twitter [1, 2, 5, 6, 34, 35], two from YouTube/Facebook (Chung et al. [36] and Salminen et al. [37]), one from Gab (Kennedy et al. [139]), and the last one was from Reddit (Karrek et al. [39]) are shown in Table 1. Finally, we merged all the multi-platform datasets into a single comprehensive dataset containing 202,377 items, and Table 2 categorizes messages as either Hate Speech or Neither.

Table 3 Optimizing various hyperparameters within a range to enhance the performance of the SSA-BiLSTM-CNN model

Hyper Parameters	Lower Bound	Upper Bound
Learning rate	0.0001	0.01
Epochs	1	50
Bi-LSTM layer	1	100
CNN layer	1	100
Dense layer	1	100

Table 4 Hyperparameter values of the SSA-BiLSTM-CNN model that remained constant

Hyper Parameters	Values
Embedding Dimension [19]	300(GLOVE)
Kernel size in Convolutional Layer	4
Dropout	0.5
Fold cross-validation	5
Loss Function	binary_crossentropy
Activation function	Relu, Sigmoid

Data Pre-Processing

Data pre-processing refers to the steps taken to manipulate, categorize, and modify data to improve the method's effectiveness. Initially, we performed data analysis to extract essential features from the textual data during the early stages of the project. The pre-processing procedure

Table 5 Iterative process of optimizing the SSA-BiLSTM-CNN model

Experiment	Learning rate	Epochs	Neurons in BiLSTM Layer	Neurons in 1D-CNN Layer	Neurons in dense layer
1	0.0084	77	92	71	98
2	0.0069	63	85	88	78
3	0.0071	56	81	85	68
4	0.0062	60	78	76	61
5	0.0046	49	74	68	58
6	0.0035	32	69	64	55
7	0.0021	20	60	62	51
8	0.0014	18	61	62	51
9	0.0014	19	60	62	51
10	0.0014	18	60	62	51

starts with removing emotional content, passwords, URLs, and various symbols such as '$', '%', and '>', as well as noise characters like ';', '&', 'abc!', '|', and ':?'. We then eliminate all stop words and convert hashtags into regular text, such as #BanRefugees becoming Ban Refugees. Next, we apply stemming, lemmatization, and capitalization to all text data. Lastly, we tokenized all standard text data, which resulted in 44,577 unique tokens from various social media datasets.

Optimizing the Hyperparameters of the SSA-Bi-LSTM-CNN Model for Optimal Performance

We have chosen five important hyperparameters of the Bi-LSTM-CNN model to be optimized, including the learning rate, epochs, number of neurons in the Bi-LSTM layer, number of neurons in the CNN layer, and number of neurons in the Dense layer. Table 3 includes a summary of each hyperparameter's range.

The BiLSTM-CNN model's parameter values that remained constant throughout the deep neural network experiment are shown in Table 4.

Results Analysis

This section examines the effectiveness of the SSA-BiLSTM-CNN model in detecting instances of hate speech. Classical models, including FNN, RNN, LSTM, CNN, and GRU, are extensively employed in hate speech detection within the field. This approach enables a direct assessment of the efficiency of the proposed model, as outlined in this paper. Furthermore, the proposed method uses the hold-out strategy to confirm optimal results. This approach randomly splits each dataset into two portions: 80% is used for training, while the remaining 20% is used for testing. Every experiment in this study has been executed using Python 3.8 and Tensorflow 2.6.0 and configured with NVIDIA Quadro K2220 32 GB on a Microsoft Windows 10 equipped with a Dual Intel Xeon E5-1650 3.50 GHz CPU and 128 GB of RAM.

Table 6 SSA-BiLSTM-CNN model hyperparameters optimization results

Hyper Parameters	Lower Bound	Upper Bound	Optimal Value
Learning rate	0.0001	0.01	0.00146
Epochs	1	100	18
Bi-LSTM layer	1	100	60
CNN layer	1	100	62
Dense layer	1	100	51

The Adam optimizer is selected to enhance the efficiency of updating the network weights. For example, the population of sparrows has been quantified as 10, whereas the percentage of individuals who have made the discovery is determined to be 20%. Optimizing all five hyperparameters of the SSA-BiLSTM-CNN model is necessary to attain the best results. It includes fine-tuning the learning rate, choosing the appropriate number of epochs, and optimizing the number of neurons in the LSTM, CNN, and dense layers. A matrix of 10×5 dimensions initializes the search range, utilizing the primary parameter settings.

The SSA algorithm aims to determine the best hyperparameter values that produce the most favourable outcomes by exploring the search space and minimizing the loss on the test set. Setting an appropriate search range for the parameters is essential for preventing issues during the search process, such as a broad search range that results in significant resource usage [41]. The study revealed that the prevalence of researchers opted for learning rates ranging from 0.001

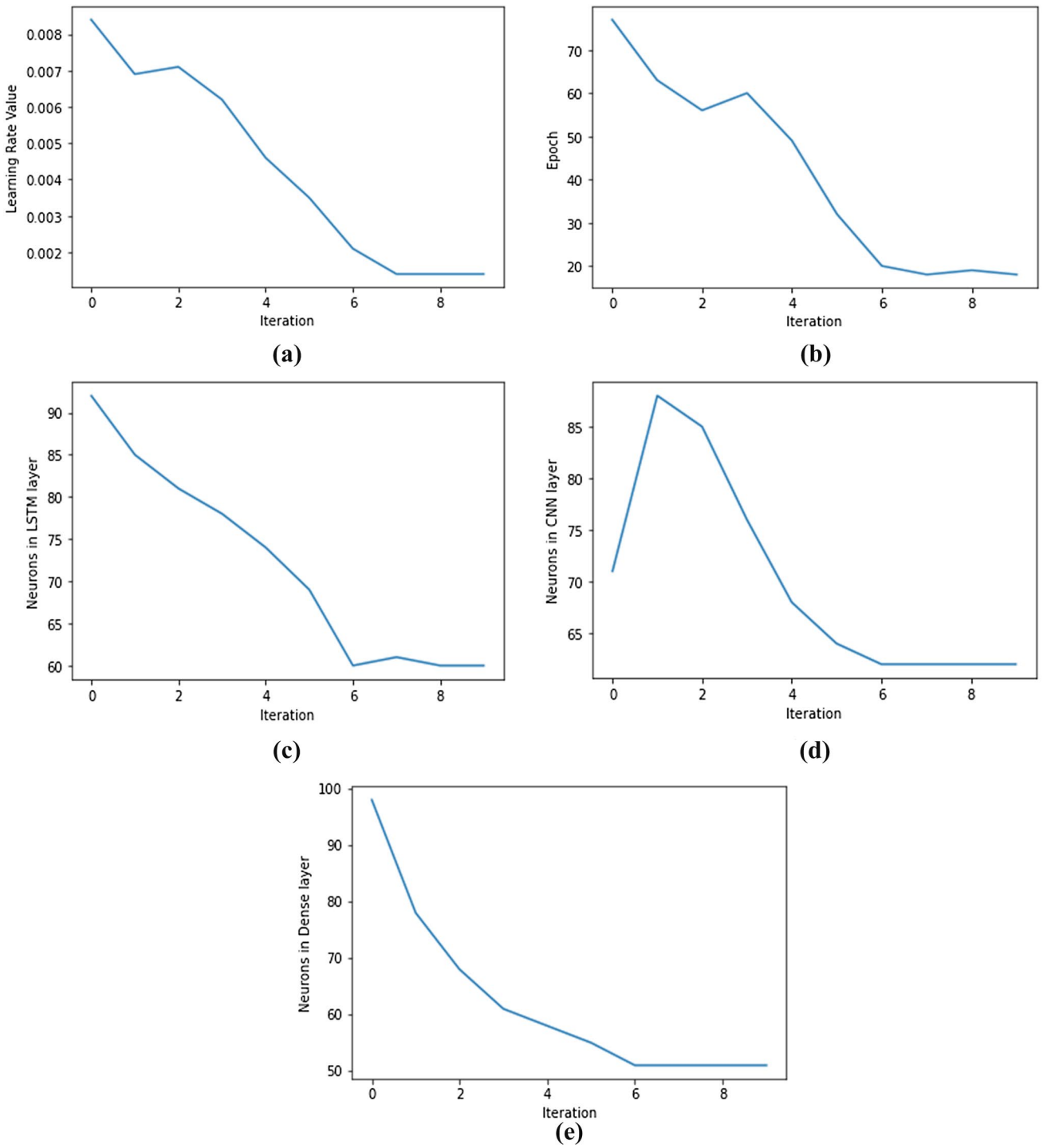

Fig. 2 The optimisation process for optimal parameters of the proposed model is depicted in Fig. 2, where **a**–**d**, and **e** represent its components. **a** Learning rate, **b** epochs, **c** Neurons in Bi-LSTM Layer, **d** Neurons in CNN Layer, and **e** Neurons in Dense layer

to 0.01, with a limited number of instances where learning rates were below 0.001 or above 0.01. Most studies that demonstrate parameter configurations maintain a limit of fewer than 100 neurons in deep neural network layers. Previous research studies have suggested using epochs within the range of 100 to approximately 200, while some have opted

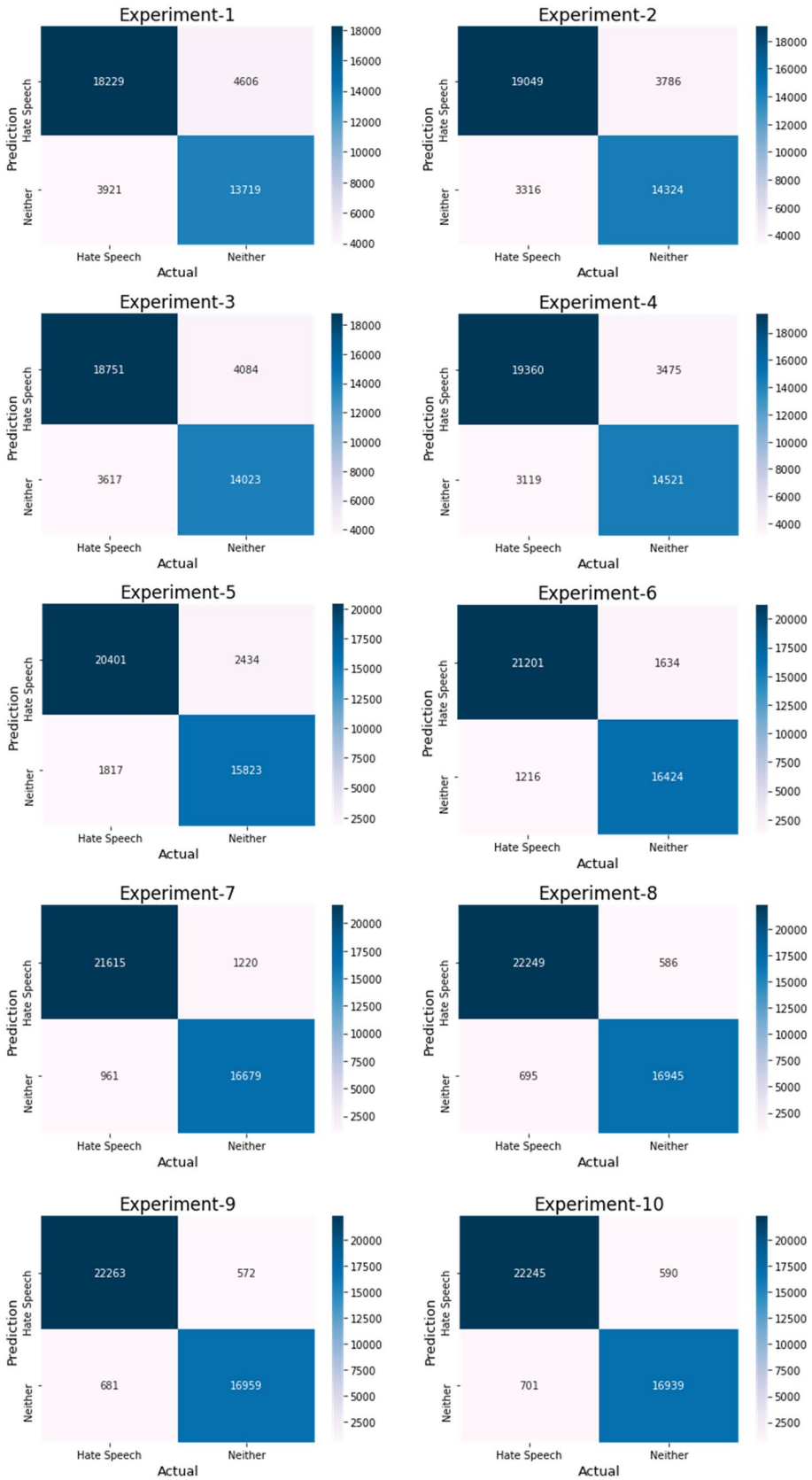

Fig. 3 Confusion matrix of SSA-BiLSTM-CNN model

Table 7 Result Analysis of the proposed SSA-BiLSTM-CNN model

Experiment	Precision	Recall	Accuracy	F1- score
1	0.798	0.822	0.789	0.810
2	0.834	0.851	0.824	0.842
3	0.822	0.835	0.809	0.828
4	0.849	0.859	0.837	0.854
5	0.893	0.918	0.894	0.905
6	0.928	0.945	0.929	0.937
7	0.946	0.957	0.946	0.951
8	0.974	0.969	0.968	0.972
9	0.974	0.970	0.969	0.972
10	**0.974**	**0.969**	**0.968**	**0.972**

Bold values indicate better results than state-of-art methods

for a more significant number of epochs. We have identified a suitable parameter search range through the analysis of these studies. The learning rate has been set to fall between 0.0001 to 0.01, and the number of neurons in the LSTM, CNN, and DENSE layers has been defined to range between 1 to 100. Our experiments showed that epochs exceeding 100 did not significantly affect the results. Therefore, we set the epoch search range to 1 to 100 due to limitations in our experimental environment.

The SSA-BiLSTM-CNN model undergoes ten experiments to optimize each parameter, as outlined in Table 5. As a result, the proposed SSA-BiLSTM-CNN model exhibits varying fitness values across each iteration. However, Table 5 demonstrates that the target parameters remained consistent throughout the seven iterations, suggesting that the SSA-BiLSTM-CNN model successfully identified the optimal parameters for the model. Furthermore, after the seven iterations, the fitness value and associated parameters exhibit stability upon attaining their optimal values.

Table 6 presents the ideal BiLSTM-CNN model parameters that the SSA determined. These parameters comprise a learning rate of 0.00146, an 18-epoch duration, 60 neurons in the BiLSTM layer, 62 in the CNN-1D layer, and 51 in the dense layer. A model for detecting hate speech is created using these optimal parameter values obtained via SSA optimization. Figure 2 illustrates the optimization process for the ideal parameters of the proposed model, with its components represented by a, b, c, d, and e. Figure 2a indicates that the learning rate's value remains consistent after reaching the optimal values, while Fig. 2b shows that the number of epochs remains unchanged after seven iterations. Also, Fig. 2c–e show the optimal values of Bi-LSTM, CNN, and Dense layer obtained by the SSA optimization.

The training set comprises 202,377 tweets classified as *"Hate Speech"* and 161,902 tweets classified as *"Neither"*. Regarding the test dataset, the number of tweets classified as *"Hate Speech"* is 22,835, whereas the number of tweets classified as *"Neither"* is 17,640. Perform the classification using these sets. Figure 3 presents an ensemble of confusion matrices accomplished by the SSA-BiLSTM-CNN model when employed on the test dataset. The data indicate that the model has successfully distinguished the text into Hate Speech and Neither category. For example, in Experiment-1, the model categorized 18,229 texts as Hate Speech and 13,719 as Neither.

Similarly, in Experiment -2, the model successfully classified 19,049 texts as Hate Speech and 14,324 as Neither. In Experiment-5, the model categorized 20,401 texts as Hate Speech and 15,823 as Neither. In Experiment -8, the model successfully identified 22,249 texts as classified under Hate Speech, while 16,945 texts have classified under the Neither class. In Experiment -9, the model categorized 22,263 texts as Hate Speech and 16,959 as Neither. Finally, the SSA-BiLSTM-CNN model has

Table 8 Comparison of proposed model with existing approaches

S.No	Paper	Data Source	ML Approach	*P*	*R*	Accuracy
1	Davidson et al. [5]	Twitter	LR, SVM, DT, NB, RF	0.866	0.865	0.865
2	Gamback et al. [40]	Twitter	Char 4 –grams, word2vec, CNN	0.857	0.721	0.783
3	Waseem et al. [16]	Twitter		0.729	0.777	0.739
4	Zhang et al. [14]	Twitter	CNN, LSTM	0.942	0.939	0.931
5	Salminen et al. [37]	Facebook, YouTube	LR, AdaBoost, DT, RF, SVM	–	–	0.789
6	Zampieri et al. [2]	Twitter	SVM, LSTM	0.824	0.821	0.802
7	Ousidhoum et al. [6]	Twitter	BiLSTM, BOW		–	0.846
8	Vashistha et al. [15]	Twitter	CNN, LSTM, BERT	0.937	0.929	0.913
9	Zhou et al. [42]	Twitter	ELMO, CNN	–	–	0.715
10	Ganfure et al. [12]	Twitter	CNN, LSTM, GRU	–	–	0.901
11	Particle Swarm Optimization (PSO)	Twitter, Facebook, YouTube, Gab, Reddit	PSO-LSTM-CNN	0.848	0.865	0.840
12	Proposed Model	Twitter, Facebook, YouTube, Gab, Reddit	SSA-BiLSTM-CNN	**0.974**	**0.969**	**0.968**

Bold values indicate better results than state-of-art methods

successfully identified 22,245 texts as Hate Speech and 16,939 texts as Neither type in experiment-10.

Table 7 presents an in-depth investigation of the classification results obtained from the SSA-BiLSTM-CNN model across ten experiments regarding precision, recall, and accuracy. The experiment's findings indicate that the model effectively classified texts across multiple iterations. In the first experiment, the model demonstrated an accuracy of 0.789. The second experiment yielded a model with an accuracy of 0.824. Likewise, the model achieved an accuracy of 0.946 in the seventh experiment. Experiment 8 resulted in a model with an accuracy of 0.968, while in Experiment 9, the model attained an accuracy of 0.969. Lastly, Experiment 10 produced a model with an accuracy of 0.968.

Table 8 compares the performance of the proposed model, SSA-BiLSTM-CNN, with those state-of-the-art methods. The model under consideration exhibited superior evaluation metrics compared to machine learning and deep learning models. First, the simulation values of the techniques developed by Davidson et al. [5], Zhang et al. [14], Salminen et al. [37], and Zampieri et al. [2] have indicated a reduced level of accuracy, with values of 0.865, 0.931, 0.789 and 0.802, respectively. Subsequently, the techniques proposed by Vashistha et al. [42] and Ganfure et al. [12] have achieved good precision, precisely 0.913 and 0.901, respectively. In addition, we also reproduce the PSO-LSTM-CNN algorithm for optimizing the proposed model to achieve less accuracy (0.840) than the SSA-BiLSTM-CNN model. This observation emphasizes the effectiveness of SSA in addressing the issue of identifying hate speech. The model's precision, recall, and accuracy under consideration are **0.974**, **0.969**, and **0.968**, respectively.

Conclusion

Detecting hate speech in textual content presents a difficulty in natural language processing. Consequently, there has been a notable increase in applying deep learning models for various natural language processing tasks. Our research introduces an innovative method to improve the optimization of hyperparameters in deep neural networks. Specifically, we have introduced an SSA-BiLSTM-CNN model for a decision support system on hate speech detection. The sparrow search algorithm has been optimized by optimizing the LSTM and CNN models' parameters. This algorithm offers an objective rationale for the model's network structure and parameter configurations. The model is trained and tested on a dataset comprising multiple social media platforms. The study compared the SSA-BiLSTM-CNN model utilizing deep learning techniques and conventional machine learning methods. The findings indicate that the SSA-BiLSTM-CNN approach attains extraordinary precision in predicting hate speech identification.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Mandl, T. et al., "Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in Indo-European languages". In Proceedings of the 11th forum for information retrieval evaluation, pp. 14–17, 2019.
2. Zampieri, M. et al.,"Predicting the type and target of offensive posts in social media, in: NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies – Proceedings of the Conference. Association for Computational Linguistics (ACL), pp. 1415–1420,2019.
3. Visualizing Eight Years of Twitter's Evolution: 2012–2019. 14 November 2019. Available online: https://blog.gdeltproject.org/visualizing-eight-years-of-twitters-evolution2012-2019/ (accessed on 12 February 2022).
4. Hochreiter S, Schmidhuber J. LSTM can solve hard long time lag problems. Advances in neural information processing systems. 1996.
5. Davidson, T. et al., "Automated hate speech detection and the problem of offensive language", in Proceedings of the 11th International Conference on Web and social media, ICWSM 2017. AAAI Press, pp. 512–515,2017.
6. Ousidhoum, N. et al." Multilingual and multi-aspect hate speech analysis". arXiv preprint arXiv:1908.11049,2019.
7. Founta, A. M., Chatzakou, D., Kourtellis, N., Blackburn, J., Vakali, A., & Leontiadis, I. (2018). A Unified Deep Learning Architecture for Abuse Detection. WebSci 2019 - Proceedings of the 11th ACM Conference on Web Science, 105–114. https://doi.org/10.48550/arxiv.1802.00385
8. Arras, L., Montavon, G., Müller, K. R., Samek, W. (2017). Explaining recurrent neural network predictions in sentiment analysis. EMNLP 2017 - 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, WASSA 743 2017 - Proceedings of the Workshop. https://doi.org/10.18653/v1/w17-5221.
9. O. Kanerva, "Evaluating explainable AI models for convolutional neural networks with proxy tasks,"2019,https://www.semanticscholar.org/paper/Evaluating-explainable-AI-models-for-convolutional_Kanerva/d91062a3e13ee034af6807e1819a9ca3051daf13.

10. Isnain, Auliya Rahman, Agus Sihabuddin, and Yohanes Suyanto. "Bidirectional long short-term memory method and Word2vec extraction approach for hate speech detection." *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)* 14.2, pp. 169–178 ,2020.
11. Zhou Y, Yang Y, Liu H, Liu X, Savage N. Deep learning-based fusion approach for hate speech detection. IEEE Access,128923–9, 2020.
12. Ganfure GO. Comparative analysis of deep learning based Afaan Oromo hate speech detection. Journal of Big Data, 2022.
13. A. Ebrahimi Fard, M. Mohammadi, Y. Chen, and B. Van de Walle, "Computational rumor detection without non-rumor: A one-class classification approach," IEEE Transactions on Computational Social Systems, vol. 6, no. 5, pp. 830–846, 2019.
14. Zhang, Z. et al., "Hate Speech Detection Using a Convolution-LSTM Based Deep Neural Network". European Semantic Web Conference, pp.745–760, 2018.
15. Pennington, J. et al., "GloVe: Global vectors for word representation", in: EMNLP 2014 – 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference.
16. Waseem, Zeerak, and Dirk Hovy. "Hateful symbols or hateful people? predictive features for hate speech detection on twitter." In *Proceedings of the NAACL student research workshop*, pp. 88–93. 2016.
17. ElSherief M, et al. Peer to peer hate: hate speech instigators and their targets. In: Proceedings of the twelfth international AAAI conference on web and social media, Palo Alto; 2018.
18. Malmasi S, Zampieri M. Challenges in discriminating profanity from hate speech. J Exp Theor Artif Intell. 2018;30:187–202.
19. Al-Ajlan, M.A.; Ykhlef, M. Optimized twitter cyberbullying detection based on deep learning. In Proceedings of the 2018 21st Saudi Computer Society National Computer Conference (NCC), Riyadh, Saudi Arabia, 25–26 April 2018; pp.1–5.
20. Ahmed, M.T.; Rahman, M.; Nur, S.; Islam, A.; Das, D. Deployment of Machine Learning and Deep Learning Algorithms in Detecting Cyberbullying in Bangla and Romanized Bangla text: A Comparative Study. In Proceedings of the 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 19–20 February 2021.
21. Dadvar, M.; Eckert, K. Cyberbullying detection in social networks using deep learning-based models. In International Conference on Big Data Analytics and Knowledge Discovery; Springer: Cham, Switzerland, 2020.
22. Luo X. Efficient English text classification using selected machine learning techniques. Alex Eng J. 2021;60:3401–9.
23. Khan U, Khan S, Rizwan A, Atteia G, Jamjoom MM, Samee NA. Aggression detection in social media from textual data using deep learning models. Appl Sci. 2022;12:5083.
24. Himdi H, Weir G, Assiri F, et al. Arabic fake news detection based on textual analysis. Arab J Sci Eng. 2022;47:10453–69. https://doi.org/10.1007/s13369-021-06449-y.
25. Alorainy W, Burnap P, Liu H, Williams ML. "the enemy among us": Detecting cyber hate speech with threats-based othering language embeddings. ACM Trans Web. 2019;13(3):1–26.
26. Sequeira R, Gayen A, Ganguly N, Dandapat SK, Chandra J. A large-scale study of the twitter follower network to characterize the spread of prescription drug abuse tweets. IEEE Trans Comput Social Syst. 2019;6(6):1232–44.
27. Zhao R, Mao K. Cyberbullying detection based on semantic-enhanced marginalized denoising auto-encoder. IEEE Trans Affect Comput. 2017;8(3):328–39.
28. Wang J, Yu L, Lai KR, Zhang X. Tree-structured regional cnn-lstm model for dimensional sentiment analysis. IEEE/ACM Trans Audio, Speech, Lang Process. 2020;28:581–91.
29. Jiankai X, Bo S. A novel swarm intelligence optimization approach: sparrow search algorithm. Syst Scie Control Eng. 2020;8(1):22–34.
30. Freitas D, Lopes LG, Morgado-Dias F. Particle swarm optimisation: a historical review up to the current developments. Entropy. 2020;22:362.
31. Liu F, Qin P, You J, Fu Y. Sparrow search algorithm-optimized long short-term memory model for stock trend prediction. Comput Intellig Neurosci. 2022;12(2022):3680419. https://doi.org/10.1155/2022/3680419.PMID:35990139;PMCID:PMC9391098.
32. Rajathi GI, Kumar RR, Ravikumar D, Joel T, Kadry S, et al. Brain tumor diagnosis using sparrow search algorithm based deep learning model. Comput Syst Sci Eng (CSSE). 2023;44(2):1793–806.
33. C. L. Zhang and S. F. Ding, "A stochastic configuration network based on chaotic sparrow search algorithm," *Knowledge-Based Systems*, vol. 220, Article ID 106924, 2021.
34. Golbeck, J. et al., "A large, labelled corpus for online harassment research". In Proceedings of the 2017 ACM on web science conference, pp.229–233,2017.
35. Fortuna, P. et al.,"A survey on automatic detection of hate speech in text". ACM Computing Surveys,2018
36. Chung, Y.L. et al., "CONAN-Counter Narratives through Nichesourcing: a Multilingual Dataset of Responses to Fight Online Hate Speech". arXiv preprint arXiv:1910.03270,2019.
37. Salminen, J. et al.," Developing an online hate classifier for multiple social media platforms. Human-centric Computing and Information Sciences,2020.
38. Kennedy B, et al. "Introducing the Gab Hate Corpus: defining and applying hate-based rhetoric to social media posts at scale. Lang Resour Eval. 2022;56(1):79–108.
39. Kurrek, J. et al., "Towards a comprehensive taxonomy and large-scale annotated corpus for online slur usage". In Proceedings of the Fourth Workshop on Online Abuse and Harms, pp. 138–149,2020.
40. Gambäck, Björn, and Utpal Kumar Sikdar. "Using convolutional neural networks to classify hate-speech." In *Proceedings of the first workshop on abusive language online*, pp. 85–90. 2017.
41. Basak R, Sural S, Ganguly N, Ghosh SK. Online public shaming on twitter: detection, analysis, and mitigation. IEEE Trans Comput Social Syst. 2019;6(2):208–20.
42. Vashistha, Neeraj, and Arkaitz Zubiaga. "Online multilingual hate speech detection: experimenting with Hindi and English social media." *Information* 12.1, 2020.