[1]Lin Yang*

Design and Performance Evaluation of Network Intrusion Detection System Based on Deep Learning

Journal of Electrical Systems

Abstract: - Today's internets are made up of nearly half million various networks. In any network connection, detecting attacks by their kinds is challenging task as various attacks may have several connections, their number vary from few to hundreds of network connections. In this paper, Design and Performance Evaluation of Network Intrusion Detection System Based on Deep Learning (NID-SPGAN-STO) is proposed. Initially, input data are collected by NSL-KDD dataset. Afterward, data are fed to preprocessing. In preprocessing, Distributed Set-Membership Fusion Filtering is used to remove redundant and biased records from input data. The pre-processed output is given to feature selection for selecting optimal features utilizing Piranha foraging Optimization Algorithm. Finally the selected features are transferred into the Semantic-Preserved Generative Adversarial Network (SPGAN) for detecting Network Intrusion, like DoS, Probe, R2L, U2R and Normal. Generally SPGAN doesn't reveal some adoption of optimization techniques for computing optimal parameters for promising precise network intrusion detection. Hence Siberian Tiger Optimisation (STO) is used to enhance weight parameters of SPGAN. The proposed NID-SPGAN-STO method is implemented using Python. To detect network intrusion detection, performance metrics likes precision, sensitivity, FI-score, specificity, accuracy, RoC, computational time are considered. The NID-SPGAN-STO method attains 30.58%, 28.73% and 25.62%, higher precision, 20.48%, 24.73%, 29.32% higher specificity and 30.98%, 26.66% and 21.32% higher F-score, 26.78%, 34.47%, and 22.86% higher recall analysed, with existing techniques likes improved binary gray wolf optimizer with SVM for intrusion detection system in WSNs (NID-SVM-IDS), network intrusion detection system utilizing deep learning (NID-DNN), design with improvement of efficient network intrusion detection system utilizing ML methods (NID-IDS-ANN) respectively.

Keywords: Distributed Set-Membership Fusion Filtering, Piranha foraging Optimization Algorithm, Semantic-Preserved Generative Adversarial Network, Siberian Tiger Optimisation.

I. INTRODUCTION

A heterogeneous system made up of tiny actuators, sensors with common-purpose computing components is called WSN [1]. WSN is made up of thousands of low-cost, self-establishing, low-power wireless node is placed across the environment for monitoring and control. While developing WSN, the following key features must be taken into account: security, robustness, scalability, self-healing, and reliability. Additionally, WSNs have a wide range of uses, including military applications, earthquake monitoring, manufacturing machine performance monitoring, and ocean monitoring [2]. Furthermore, WSN principles are probably going to used in architectures of future applications comprising building security, wildfire prevention, pollution monitoring, traffic monitoring, and water quality monitoring [3]. The conversion of raw data into helpful aggregated and grouped information is just one of the numerous benefits of WSNs [4]. In addition to being an excellent data processing, storing facility, human interface access point, BS usually serves as gateway to another network [5]. Moreover, can serve as connector to pull data from network, distribute control information [6]. Additionally, the base station has mentioned the washbasin. All BS serves as root of routing forest that is formed by each sensor nodes [7].Usually, the base station has more memory to store cryptographic keys, additional powerful (faster and more powerful) CPUs, a way to connect with other WSNs, and enough battery life to last lifetime of sensor nodes [8]. Consequently, it is best to utilise IDS to find unusual activity, intrusions. The process of choosing the most relevant features to employ in the construction of reliable and accurate IDS models is known as feature selection, or FS [9]. FS's primary goals are to reduce dimensionality of data, enhance the performance of identified performance. Numerous aspects, including redundant ones, are employed in data representation in real-world applications. Consequently, certain features would assume duty of other features, additional features could be divided. Furthermore, output is directly affected by the pertinent features, and they contain crucial data that characterises the dataset's behaviour [10]. A thorough search for optimal features set in higher-dimensional space was previously impractical. General performance of prediction scheme depend on how well it forecasts unknown classes regards accuracy, total selected features, how quickly it executes the changes [11]. It improves performance of GWO-IDS in WSN with increase wolves, utilizing multi-objective function. To maximise classification accuracy while using the fewest possible features, these two binary GWO approaches are frequently applied in the FS domain [12].

[1]* *Department of Information, Hebei Youth Administrative Cadres College, Shijiazhuang, Hebei, 050031, China*

**Corresponding author e-mail: yanglin840302@163.com*

The performance of suggested method is tested using NSL KDD'99 dataset, its outcomes are compared with those of other methods currently in use, such as PSO-IDS (due to its lengthy execution time, lower detection rate), GWOSVM-IDS with three wolves (due to its lengthy execution time, lower accuracy). The accuracy, execution time, feature count, false alarm rate, detection rate of suggested approaches are assessed [13].

Furthermore, performance of detection process is impacted by higher degree of categorization, higher complexity, lengthy processing time, large storage capacity. As a result, improving classifier performance and reducing processing costs continue to be the primary problems with IDS and require more work. But decreasing the dataset's dimensionality would improve the detection process' efficiency and solve the overhead classification problem.

Major contributions of this manuscript brief as below:

- The proposed NID-SPGAN-STO system contributes to the field based Design and Performance Evaluation of Network IDS depend on DL for NID.
- Utilization of NSL-KDD dataset as input data for training, testing the system is a valuable for assessing the effectiveness of proposed system.
- By utilizing the Piranha foraging Optimization Algorithm (PFOA), based feature selection the feature from higher dimension to lower dimension without losing important information.
- Semantic-Preserved Generative Adversarial Network utilized for detecting Network Intrusion, like DoS, Probe, R2L, U2R and Normal.

Remaining part of this paper arranged as below: segment 2 examines literature review, proposed method defined in segment 3, results with discussion is established in segment 4, conclusion.

II. LITERATURE REVIEW

Numerous investigation studies were presented in literature connected to Network IDS utilizing DL; some of current investigations were assessed in this part.

Safaldin et al. [14] has presented improved binary gray wolf optimizer and SVM for IDS in WSNs. Here, Using two distinct methods, three binary enhanced variants of GWO are presented. The first strategy involves the individual identifying the first three best options, binarizing them, and then performing a stochastic crossover between the three main moves to determine where the binary grey wolf will next appear. In the second method, a sigmoidal function is used to update the location of the wolves frequently. The presented method provides higher precision, but it gives lower computation time.

Ashiku and Dagli [15] have presented Network IDS utilizing DL. Here, proposes the creation of robust, adaptable network IDS that uses deep learning architectures to detect, categorise network threats. Based on DL-DNNs might enable adaptive IDS that can identify and eliminate unknown or zero-day network behavioural characteristics, hence expelling system intruders and mitigating compromise risk. It attains higher RoC, lower specificity.

Rincy and Gupta [16] has presented Design and development of effectual network IDS utilizing ML methods. An effective hybrid NID-Shield NIDS was described. The UNSW-NB15 and NSL-KDD 20% dataset were categorised by hybrid NID-Shield NIDS based on attack names and kinds. Some attacks, like R2L, U2R, have relatively some N/W connections, but other attacks, like DoS, probe, many N/W connections, or they grouping of some of them. Different assaults may have strange connections. It provides high f-measure, lower precision.

Imran et al. [17] has presented intelligent and effectual network IDS utilizing DL. Here, offers fresh method for detecting network intrusions that is based on the DL technique. Included a discussion of the problems in the NIDS methods as they exist.To create a novel approach using the SVM classification algorithm and stacked NDAE. Tensor-Flow was used, and a thorough assessment of its capabilities was conducted. It attains higher precision, lower computational time.

Kunang et al. [18] has presented attack classification of IDS utilizing DL and hyperparameter optimization. Here, provides an alternate method of choosing DL structure models combines grid search, random search methods to automatically carry out the HPO procedure. The HPO procedure looks for predicted hyperparameter candidate values that were most probable to improve DL method's classification in IDS. It provides high sensitivity and low accuracy.

Sohi et al. [19] has presented improving network IDS through DL. Here, suggests that RNNs can be used to create novel, formerly undiscovered attack variants, artificial signatures from most sophisticated malware to increase intrusion detection rate. Additionally, RNNs can be used to create malicious datasets that contain, for

example, malware variants that have not yet been discovered in order to improve the design of an NIDS. It provides high f-score and low RoC.

Gowdhaman and Dhanapal [20] has presented IDS for WSNs utilizing DNN. Here, IDS depend on DL. Although detection accuracy, other performance metrics improved, IDS still make wide utilize of earlier ML approaches. When analysed with traditional ML techniques, suggested IDS executes better overall. It attains higher precision, lower recall.

III. PROPOSED METHODOLOGY

The NID-SPGAN-STO is discussed. It presents Design and Performance Evaluation of Network IDS depend on DL. Block diagram of NID-SPGAN-STO is represented by Figure 1. Thus, the detailed description about NID-SPGAN-STO is given below,

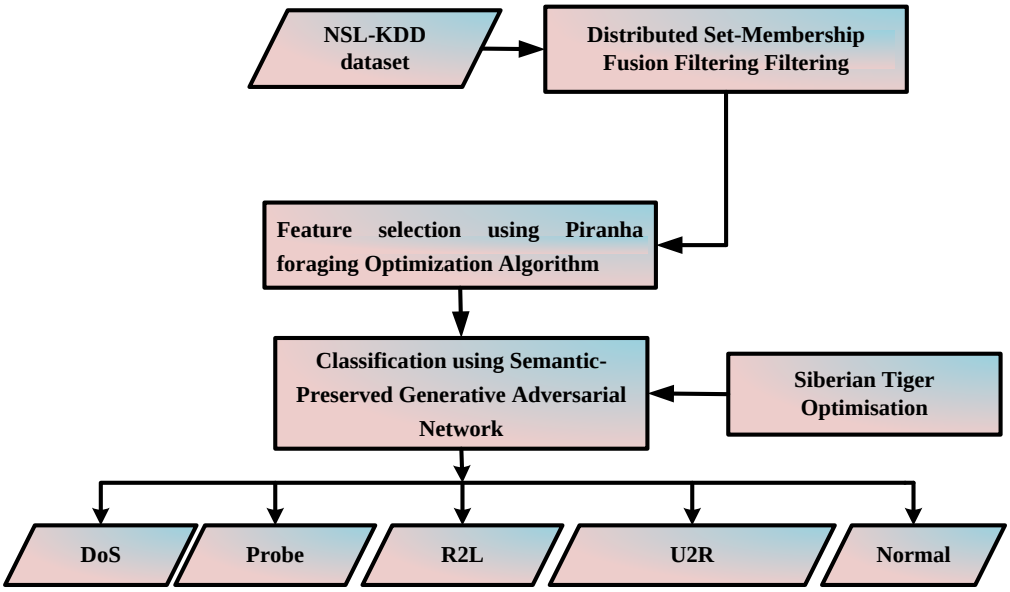

Figure 1 Block diagram of NID-SPGAN-STO methodology

A. Data acquisition

The data are gathered from NSL-KDD dataset [21]. It collected via NSL-KDD dataset to categorize attack kinds. Four classes of attacks occur in NSL-KDD dataset likes Probe, U2R, R2L, DoS, Normal. Table 1 represents features of NSL-KDD. DoS: Targeting memory resources. Remote to Local: These types of attacker of particular node without recognition to neighbouring legitimate nodes and forward packets. User to Root: exploit specific risks for super users who are authorized to use valid nodes. Probe: collect data to create numerous safety threats of whole Network.

Table 1: NSL-KDD Dataset Features.

No. of features	Name of feature	No. of features	Name of feature
1	duration	21	Dtcpb
2	'protocol_type	22	'num_outbound_cmds'
3	service	23	is_host_login'
4	flag	24	is_guest_login
5	src_bytes	25	count
6	dst_bytes	26	srv_count
7	land	27	serror_rate

8	wrong_fragment	28	srv_serror_rate
9	urgent	29	rerror_rate
10	hot	30	srv_rerror_rate
11	Logged_in	31	diff_srv_rate
12	num_compromised	32	srv_diff_host_rate
13	root_shell	33	same_srv_rate
14	su_attempted	34	dst_host_count
15	num_root	35	dst_host_srv_count
16	num_file_creations	36	dst_host_same_srv_rate
17	num_shells	37	dst_host_diff_srv_rate
18	num_access_files	38	dst_host_same_src_port_rate
19	'dst_host_srv_diff_host_rate	39	dst_host_srv_serror_rate
20	dst_host_serror_rate	40	Class

B. Pre-processing using Distributed Set-Membership Fusion Filtering

In this step, DSMFF [22] is used for remove redundant and biased records from input data. DSMFF technique allows individual nodes in a network to independently process and filter their own local measurements and estimates. These nodes then collaborate and exchange information with neighbouring nodes to achieve a collective estimate of the system state. Frame vector is passed to channel coder, extracts data before passing it to the channel accumulator, aggregates channel data from each frames that are presented in equation (1),

$$\hat{w}^{e}_{b+1,c+1} = x^{(1)}\left(\hat{w}^{e}_{c+1}\right) + d^{(2)}\left(w^{(e)}_{b+1,c}\right) + F^{(e,1)}_{b,c+1} + G_{b.c+1}{}^{e} + F^{(e,2)}_{b,c+1} + G_{b.c+1}{}^{e} \quad (1)$$

Here $\hat{w}^{e}_{b,c} \supseteq H^{l_a}$ denotes local estimate of $w_{b,c}$ on sensor e, $F^{e,1}_{b,c}$, $F^{e,2}_{b,c}$ b,c denotes filter parameters to be calculated. Functions comprise two or more rectangular regions enclosed in template. Feature value of distributed set-membership fusion filtering with - rectangles are obtained using equation (2),

$$X^{(1)}\left(w_{b,c}\right) = x^{(1)}\left(\hat{w}^{e}_{b,c}\right) + \Omega^{(e,1)}_{b,c} \quad (2)$$

Where $\Omega^{(e,1)}_{b,c}$ denotes Jacobian matrices, and $X^{(p)}\left(w_{b,c}\right)$ signifies for remove redundant and biased records from input data using equation (3),

$$\varphi^{e}_{j,i} = \int w_{b,c}(b,x) \tag{3}$$

Where j,i denotes mean intensity of pixel in image enclosed by $j-th$ rectangle by mean value of φ .Lastly, by normalising data, pre-processed Distributed Set-Membership Fusion Filtering approach removed redundant and biased records from the input data. Then, pre-processed output is given into feature extraction phase.

C. Feature selection utilizing Piranha foraging Optimization Algorithm

The feature selection using PFOA [23] is discussed. The PFOA visualization is employed to evaluate effectiveness of PFOA optimization, analyze features of three foraging modes, determine parameters sensitivity, determine size of piranha population on process. The process presentation was established, four real engineering design optimization issues, outcomes were likened by 13 known meta-heuristics.

1) Stepwise process of PFOA

The stepwise procedure is defines choosing optimal feature selection using Hybrid PFOA Algorithm. Initially, PFOA Algorithm creates initial, uniformly distributed population to select optimal features. It is stimulated using PFOA Algorithm. The procedure of detailed step is depicted here,

Step 1: Initialization

PFOA procedure describes three patterns of localized group attack, bloodthirsty cluster attack, scavenging foraging, its steps comprise primarily of population initialization, population assessment, parameter and agent location apprising in equation (4),

$$W_j = va_j + \alpha_1 \times (Wa_j - va_j) \tag{4}$$

Where W_j denotes location of $j-th$ individual piranha candidate solution, va_j signify upper, lower boundaries of piranha search in habitat.

Step 2: Random Generation

Features present in pre-processed dataset are selected randomly utilizing the aid of PFOA approaches.

Step 3: Fitness Function

From initialized evaluations, random solution is created. It is given in equation (5).

$$Fitness\ function = [selecting\ optimal\ features] \tag{5}$$

Step 4: Define predation intensity parameter

Piranhas are very sensitive to detection of blood, typical influenced by blood concentration H_j , distance G_j among prey, piranha denoted in equation (6),

$$G_j = \alpha_2 \times \frac{H_j}{4\lambda c_j^2} \tag{6}$$

Where G_j means predation intensity parameter for position of $j-th$ individual piranha, λc signifies distance among location.

Step 5: Non-linear parametric control strategies

Nonlinear parametric control approaches are effectual scales utilized to control time-varying randomized procedures, prevent premature convergence of populations though ensure smooth with silky transition is shown in equation (7),

$$R = D.\cos\left[\frac{\lambda}{2} \otimes \left(\frac{e}{Maz\_jter}\right)\right]^4 \tag{7}$$

Where Maz_jter signifies maximum iterations, constant D is proved, $\otimes$ signifies product of value, variable.

Step 6: Termination Condition

Optimal feature is choose depend on PFOA Algorithm iteratively repeat step 4 till fulfil $W = W + 1$ halting conditions. Such are given input for IDS. If all process are achieved it will select the accurate feature for attaining better Network Intrusion Detection. Table 2 depicts selected features using PFOA

Table 2: Selected features using PFOA

No.of features	Name of feature	No.of features	Name of feature
1	su_attempted',	6	'dst_host_srv_diff_host_rate'
2	num_root'	7	'dst_host_diff_srv_rate'
3	num_file_creations'	8	'dst_host_same_src_port_rate'
4	num_shells'	9	'dst_host_srv_serror_rate'
5	'num_access_files'	10	'class'

D. Network Intrusion detecting utilizing Semantic-Preserved Generative Adversarial Network

In this section, Network Intrusion detecting using SPGAN [24] is discussed. SP-GANs domain adaptation, where method trained on source domain is transferred to a target domain with different data distribution. By generating synthetic data in the target domain, SP-GANs can help bridge the gap between source, target domains, improving method's performance in target domain. The preservation of semantic features facilitates better transferability of knowledge between real and DoS is presented in equation (8),

$$V_{GAN}\left(F,G,C_R,C_E,J_{R,}J_E\right)=T_{y_e\sim}W_E\left[\log C_E\left(y_e\right)\right] \tag{8}$$

To promote the preservation of the original material throughout the conversion process, apply a Probe constraint. Next, in order to enforce cycle consistency, the original sample should be duplicated by mapping source sample from source to target, probe as follows in equation (9),

$$V_{imgf}\left(F,G,J_R,J_E\right)=T_{yr\sim}Y_R\left[\| G_{img}\left(w_r\right)-w_r \|_1\right] \tag{9}$$

Where V_{imgf} stands for ℓ1 normal R2L is the result of GNA-depend source-domain-target-domain image translation techniques that solely take into account the two losses stated above. In the end, the generator's images have a distribution that is far closer to the R2L, is preserved in equation (10),

$$V_{rep}\left(F,G,J_{R,}J_E\right)=T_{w_r}\sim W_R\left[\| F_{enc}\left(w_s\right)-G_{enc}\right] \tag{10}$$

On the other hand, this strategy doesn't necessitate some pre-trained methods in U2R, allows for the efficient and U2R introduction of semantic constraints during the training process is follows in equation (11),

$$V_{sem}\left(F,J_R\right)=T_{\left(w_{r,}x_r\right)\sim\left(W_{R,}X_R\right)}\left[v\left(F_{sem}\left(w_r\right),w_r\right)\right] \tag{11}$$

Where V_{sem} indicates the cross-entropy loss function and w_r denotes label of source domain, V_{sem} denotes probability. Along with above term loss, general loss function of SPGAN designed Normal in equation (12),

$$V_{SPGAN}=V_{GAN}\left(F,G,C_R,C_E,J_R,J_E\right)+\gamma_1 V_{ing}\left(F,G,J_{R,}J_E\right) \tag{12}$$

Where γ_1 denotes typically set to value within γ_2 is classically set to 1, γ_3 is increasingly Normal. Properly, given labelled translated image G ,unlabelled target image then mixed image formed in Normal using equation (13),

$$w_n=N_{mask}\otimes y_r'+\left(1-N_{mask}\right)\otimes y_e \tag{13}$$

Where $\otimes$ denotes element-wise products. To become labels of mixed image w_n , Mean teacher method engaged to allocate pseudo-labels to target data. Translated image boundary loss given in equation (14),

$$V_{total}=V_{seg}+\gamma_a.\left(V_{src\_a}+V_{mix\_a}\right) \tag{14}$$

Where γ_a denotes hyper-parameter to control weight boundary improvement module. Finally, the SPGAN for detecting Network Intrusion, like DoS, Probe, R2L, U2R and Normal. In this work, Siberian Tiger Optimization Algorithm process exploited for enhancing optimum parameters of SPGAN classifier. The weight and bias parameters of the SPGAN are adjusted using STO. For constraint generation, methods including grid exploration, manual exploration, random exploration are usually utilized. Nonetheless, such investigations exhibit a peculiar weakness in terms of repetition time, and there isn't any familiar research produced through deceit.

E. Optimization using Siberian Tiger Optimization Algorithm

The weights parameter $V\,and\,W$ of SPGAN is enhanced utilizing STO [25]. STO is a nature-inspired optimization process that mimics hunting behaviour and social interactions of Siberian tigers. It was proposed based on the belief that tigers possess effective strategies for seeking prey and avoiding threats in their natural environment.

1) Stepwise process of STO

The stepwise procedure is defines attain optimal values of SPGAN utilizing STO. Initially, STO makes the uniformly distributed population for enhancing optimal parameters of SPGAN parameters. Optimal solution is promoted utilizing STO method then equivalent flowchart is present in figure 2. Procedure of complete stage is present as following,

Step 1: Initialization

Siberian tigers comprising the STO population traverse the search space in pursuit of more comprehensive solutions. All Siberian tigers are members of the STO population. It is denoted in equation (15),

$$F=\begin{bmatrix} f_{1,1} & \cdots & f_{1,u} & \cdots & f_{1,f} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ f_{v,1} & \cdots & e_{v,u} & \cdots & f_{v,f} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ f_{M,1} & \cdots & f_{M,u} & \cdots & f_{M,f} \end{bmatrix} \tag{15}$$

Where, e denotes population matrix of Siberian tigers' positions, F_v signifies v^{th} Siberian tiger, M signifies total number of Siberian tigers.

Step 2: Random Generation

The input parameters are made at randomly. Explicit hyper parameter condition determines the optimal fitness value selection.

Step 3: Fitness function estimation

Initialized assessments utilized to made random solution. It is evaluated by parameter optimization value enhancing weight parameter $V\,and\,W$ of the classifier. It given in equation (16),

$$fitness\ \ function = optimizing\ (V\,and\,W) \tag{16}$$

Step 4: Exploration phase for Prey Hunting

Proposed prey placements for each Siberian tiger in the STO design come from other population members are more likely to survive and provide a more useful purpose than that particular member. It is given in equation (17),

$$E_v=\begin{cases} E_u^{K1O1}, & C_u^{K1O1}<C_v; \\ E_v, & else \end{cases} \tag{17}$$

Where, E_v^{K1O1} denotes objective function value of V^{th} member C_v^{K1O1}. The Siberian tiger roams the area, attacking its prey in the process. The algorithm's ability to perform local exploration and exploitation and find the optimum answer is enhanced by this process.

Step 5: Exploitation phase for optimizing $V\,and\,W$

In the second step, the population's geographical placement is altered through a battle simulation. Owing to this property, STO's local search and exploitation capabilities are enhanced by slight movements in population members' placements. In order to replicate this behaviour, an equation is first used to generate a random point close to the fight area. It is given in equation (18),

$$V_{v,u}^{K2O2} = e_{v,u} + \frac{b_{v,u}}{z}\left(Wr_u - sr_u\right)$$

(18)

Where, $V_{v,}^{K2O2}$ denotes new location of v^{th} Siberian tiger depend on second stage of 2nd phase of STO, $_{v,u}^{K2O2}$ signifies u^{th} dimension, W denotes iteration counter.

Step 6: Termination condition

The weight parameter values of generator V *and* W from deeply supervised shuffle attention convolutional neural network are optimized by support of STO process, repeat step 3 until halting conditions $F = F + 1$ is met**.** Next, SPGAN detects network intrusions of various kinds, including DoS, Probe, R2L, U2R, and Normal. More accuracy through error-free calculating time reduction.

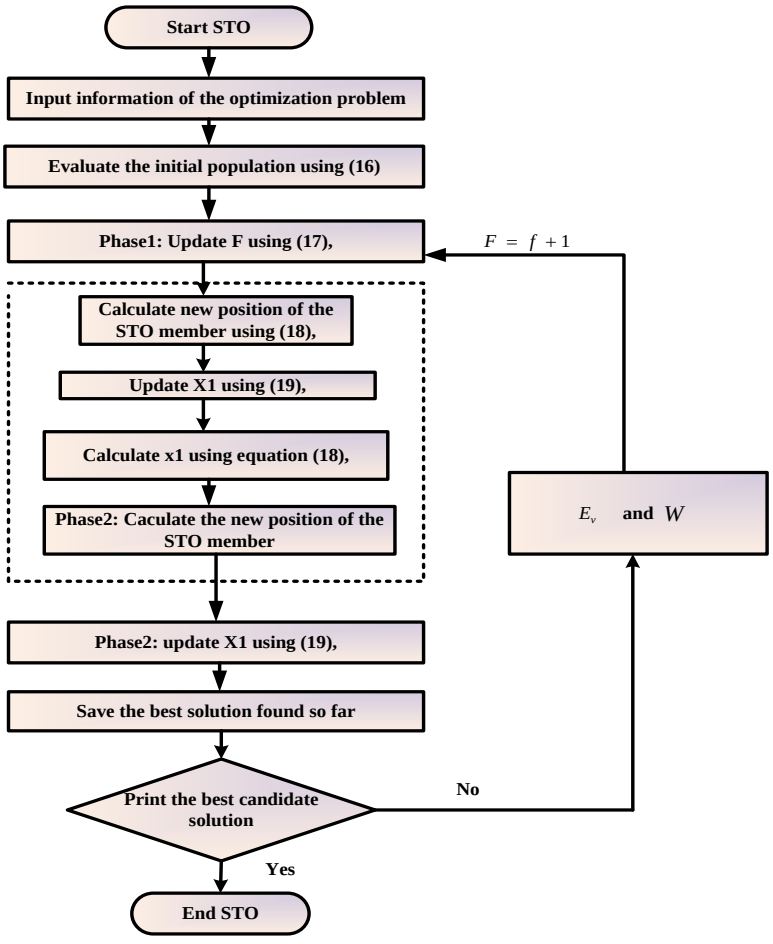

Figure 2: Flowchart of STO for enhancing SPGAN parameter

IV. RESULT AND DISCUSSION

Experimental results of suggested technique are deliberated. The proposed method is implemented in python. Utilising a workstation equipped by 11-GEN CPU and Intel Core i7 by 8 GB RAM. Obtained outcomes of NID-SPGAN-STO technique is analysed with existing technique likes NID-SVM-IDS, NID-DNN, NID-IDS-ANN respectively.

A. Performance measures

Selecting the best classifier requires taking this critical step. Accuracy, precision, sensitivity, specificity, FI-score, error rate, ROC, computational time is among performance metrics in order to evaluate performance. Performance metric is deemed in order to scale the metrics. True Negative, True Positive, False Negative and False Positive are necessary to scale performance metric.

1) Accuracy

The degree of agreement among measured or observed value actual or accepted value of quantity is known as accuracy. It determined by equation (19),

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)} \tag{19}$$

Here, TP denotes True positive, TN denotes True negative, FN denotes False negative, FP denotes False positive.

2) Precision

The degree of consistency or reproducibility in obtaining the same results in measurements under comparable settings is referred to as precision. Through the use of equation (20)

$$\text{Pr}ecision = \frac{TP}{(TP + FP)} \tag{20}$$

3) F-score

F-score mean accuracy and Precision are also parts of the weighted. Equation (21) was used to calculate.

$$F - Scorevalue = 2 \times \frac{recall \times precision}{recall + precision} \tag{21}$$

4) Specificity

There is an actual negative rate in it. It is given in equation (22),

$$\text{Specificity} = \frac{TN}{(FP + TN)} \tag{22}$$

5) RoC

RoC is graphical representation that illustrates trade-off among true and false positive rate for binary classification system at various thresholds. It scaled in equation (23),

$$ROC = 0.5 \times \left(\frac{TP}{TP + FN} + \frac{TN}{TN + TP} \right) \tag{23}$$

6) Recall

Recall is considered by separating total elements in positive class by number of genuine positives. It's determined by equation (24),

$$\text{Re}call = \frac{TP}{(TP + FN)} \tag{24}$$

7) Error rate analysis

It is usually stated as ratio of mistakenly predicted cases to total examples evaluated, is a statistic used to quantify the frequency of incorrect predictions made by a model. It analysis are shown in equation (25),

$$Errorrate = 100 - Accuracy \tag{25}$$

B. Performance Analysis

Fig 3 to 9 displays simulation outcomes of NID-SPGAN-STO technique. Then, the proposed OSD-PRGAN-SAR method is likened with existing NID-SVM-IDS, NID-DNN, and NID-IDS-ANN methods respectively.

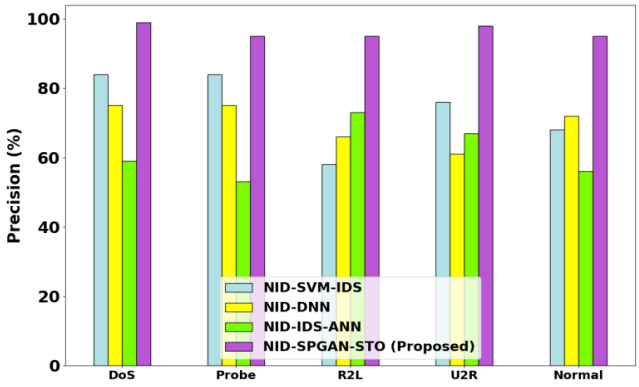

Fig 3: Precision analysis

Figure 3 portrays precision analysis. A statistical parameter called precision is utilized to evaluate classification method's accuracy; it is especially useful for analysing binary classification situations. The proposed NID-SPGAN-STO method attains 78.13%, 74.53% and 82.92%, higher precision for DoS; 60.83%, 62.46% and 83.98% greater precision for Probe; 70.83%, 68.46% and 78.98% greater precision for R2L; 67.83%, 72.46% and 80.98% greater precision for U2R; 70.83%, 72.46% and 81.98% greater precision for normal with existing techniques likes NID-SVM-IDS, NID-DNN, and NID-IDS-ANN methods respectively.

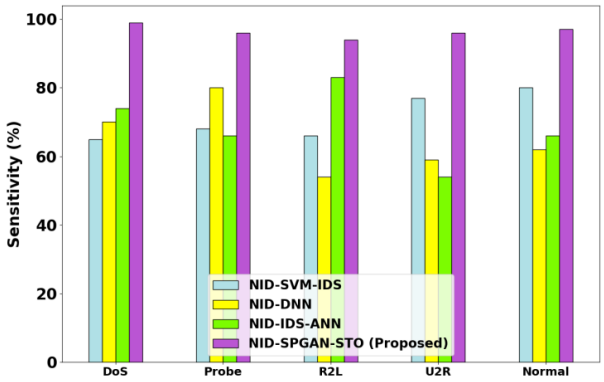

Fig 4: Sensitivity analysis

Figure 4 depicts Sensitivity analysis. Sensitivity, which is defines ratio of true positives to false negatives, assesses how well a classification model can identify positive instances among all actual positives. The proposed NID-SPGAN-STO method attains 75.13%, 72.53% and 80.92%, higher Sensitivity for DoS; 70.83%, 72.46% and 73.98% greater Sensitivity for Probe; 80.83%, 78.46% and 79.98% greater Sensitivity for R2L; 60.83%, 72.46% and 83.98% greater Sensitivity for U2R; 75.83%, 78.46% and 81.68% greater Sensitivity for normal with existing techniques likes NID-SVM-IDS, NID-DNN, and NID-IDS-ANN methods respectively.

Figure 5 depicts precision F-score. The F-score, which is determined by taking harmonic mean of precision, recall, is statistic merges precision, recall into single value, balancing the trade-off between both. The proposed NID-SPGAN-STO method attains 65.13%, 62.53% and 70.92%, higher F-score for DoS; 74.83%, 70.46% and 73.98% greater F-score for Probe; 88.83%, 74.46% and 79.98% greater F-score for R2L; 67.83%, 72.46% and 73.98% greater F-score for U2R; 75.83%, 78.46% and 81.68% greater F-score for normal with existing techniques likes NID-SVM-IDS, NID-DNN, and NID-IDS-ANN methods respectively.

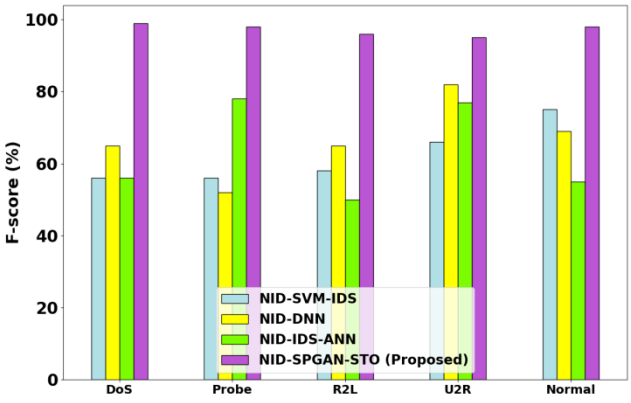

Fig 5: Analysis of F-score

Figure 6 depicts Specificity analysis. Specificity measures a classification method's ability to correctly detect negative instances among total actual negatives, expressed as ratio of true negatives and false positives. The NID-SPGAN-STO method attains 65.13%, 62.53% and 70.92%, higher Specificity for DoS; 84.83%, 80.46% and 77.98% greater Specificity for Probe; 81.83%, 74.46% and 79.98% greater Specificity for R2L; 68.83%, 72.46% and 79.98% greater Specificity for U2R; 75.83%, 78.46% and 81.68% greater Specificity for normal with existing techniques likes NID-SVM-IDS, NID-DNN, and NID-IDS-ANN methods respectively.

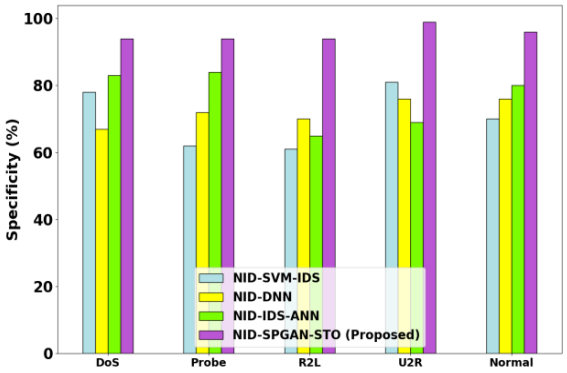

Fig 6: Specificity analysis

Figure 7 depicts accuracy analysis. A classification model's overall correctness is gauged by its accuracy, which is resolute by dividing cases correctly forecast total instances. The proposed NID-SPGAN-STO method attains 65.13%, 62.53% and 70.92%, higher accuracy for DoS; 84.83%, 80.46% and 77.98% greater accuracy for Probe; 81.83%, 74.46% and 79.98% greater accuracy for R2L; 68.83%, 72.46% and 79.98% greater accuracy for U2R; 75.83%, 78.46% and 81.68% greater accuracy for normal with existing techniques likes NID-SVM-IDS, NID-DNN, and NID-IDS-ANN methods respectively.

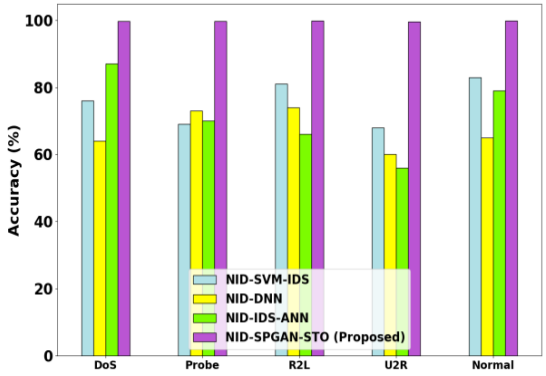

Fig 7: Accuracy Analysis

Figure 8 depicts RoC analysis. Receiver Operating Characteristic is graphical representation of classification method's performance across different discrimination thresholds, plotting trade-off among true and false positive rate. The NID-SPGAN-STO technique attains 0.91%, 0.93% and 0.96% higher ROC analysis analysed with existing methods such as NID-SVM-IDS, NID-DNN, and NID-IDS-ANN respectively.

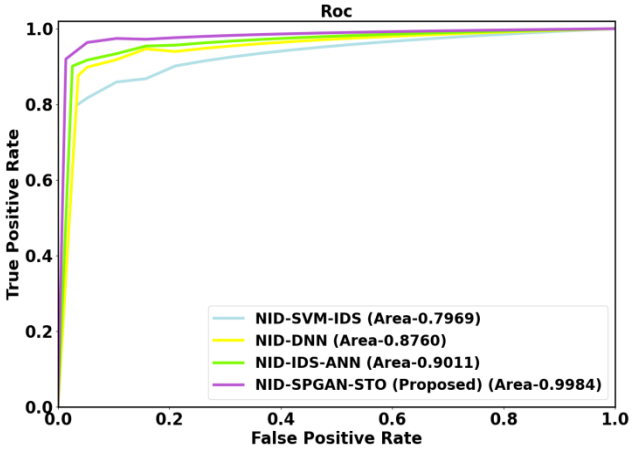

Fig 8: ROC analysis

Figure 9 displays computation time analysis. Computation time is the amount of time, usually expressed in seconds, minutes, or other time units, that a computer needs to do a certain activity or procedure. Here, NID-SPGAN-STO attains 36.81%, 40.41% and 37.75% lesser computation time analysed with existing method likes NID-SVM-IDS, NID-DNN, and NID-IDS-ANN models respectively.

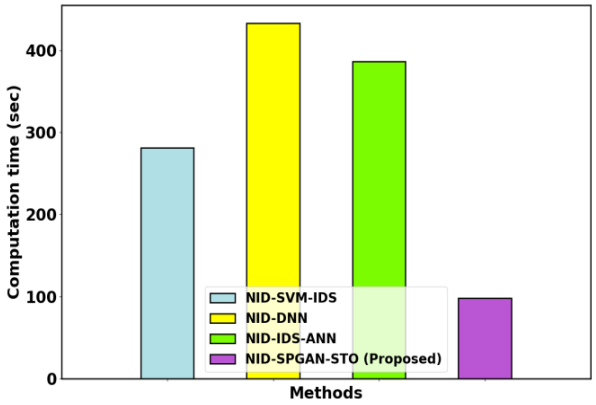

Figure 9: Computation Time analysis

C. Discussion

In the proposed, the complexity of identifying network attacks within the vast landscape of nearly half million different networks is acknowledged. Addressing this challenge, the authors introduce the Design and Performance Evaluation of Network IDS depend on DL (NID-SPGAN-STO). Leveraging NSL-KDD dataset, the input data undergoes a comprehensive preprocessing stage, employing Distributed Set-Membership Fusion Filtering to eliminate redundant and biased records. Then, to find the best features, feature selection is carried out using the Piranha foraging Optimisation Algorithm. The Semantic-Preserved Generative Adversarial Network (SPGAN) is equipped with the chosen features to identify many kinds of network intrusions, comprising DoS, Probe, R2L, U2R, Normal. The highlight, in particular, is the novel use of Siberian Tiger Optimisation (STO) to optimise the weight parameters of SPGAN a strategy not frequently found in other SPGAN-based intrusion detection systems. The implementation, conducted in Python, is evaluated utilizing performance metrics likes precision, sensitivity, FI-score, specificity, accuracy, RoC, computational time.

V. CONCLUSION

In this section, NID-SPGAN-STO was successfully implemented. The proposed NID-SPGAN-STO method attains 30.58%, 28.73% and 25.62%, higher RoC, 20.48%, 24.73%, 29.32% higher computational time and 30.98%, 26.66% and 21.32% higher Accuracy analysed, with existing techniques like NID-SVM-IDS, NID-DNN, and NID-IDS-ANN respectively.

REFERENCE

[1] Sstla, V., Kolli, V.K., Voggu, L.K., Bhavanam, R., & Vallabhasoyula, S. (2020). Predictive Model for Network Intrusion Detection System Using Deep Learning. *Revue d'Intelligence Artificielle*, *34*(3).

[2] Alavizadeh, H., Alavizadeh, H., & Jang-Jaccard, J. (2022). Deep Q-learning based reinforcement learning approach for network intrusion detection. *Computers*, *11*(3), 41.

[3] Lee, S.W., Mohammadi, M., Rashidi, S., Rahmani, A.M., Masdari, M., & Hosseinzadeh, M. (2021). Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review. *Journal of Network and Computer Applications*, *187*, 103111.

[4] Alhajjar, E., Maxwell, P., & Bastian, N. (2021). Adversarial machine learning in network intrusion detection systems. *Expert Systems with Applications*, *186*, 115782.

[5] Saba, T., Rehman, A., Sadad, T., Kolivand, H., & Bahaj, S.A. (2022). Anomaly-based intrusion detection system for IoT networks through deep learning model. *Computers and Electrical Engineering*, *99*, 107810.

[6] Binbusayyis, A., & Vaiyapuri, T. (2021). Unsupervised deep learning approach for network intrusion detection combining convolutional autoencoder and one-class SVM. *Applied Intelligence*, *51*(10), 7094-7108.

[7] Chen, D., Yan, Q., Wu, C., & Zhao, J. (2021). Sql injection attack detection and prevention techniques using deep learning. *In Journal of Physics: Conference Series* (Vol. 1757, No. 1, p. 012055). IOP Publishing.

[8] Mighan, S.N., & Kahani, M. (2021). A novel scalable intrusion detection system based on deep learning. *International Journal of Information Security*, *20*, 387-403.

[9] Bertoli, G.D.C., Júnior, L.A.P., Saotome, O., Dos Santos, A.L., Verri, F.A.N., Marcondes, C.A.C., Barbieri, S., Rodrigues, M.S., & De Oliveira, J.M.P. (2021). An end-to-end framework for machine learning-based network intrusion detection system. *IEEE Access*, *9*, 106790-106805.

[10] Lansky, J., Ali, S., Mohammadi, M., Majeed, M.K., Karim, S.H.T., Rashidi, S., Hosseinzadeh, M., & Rahmani, A.M. (2021). Deep learning-based intrusion detection systems: a systematic review. *IEEE Access*, *9*, 101574-101599.

[11] Zhong, M., Zhou, Y., & Chen, G. (2021). Sequential model based intrusion detection system for IoT servers using deep learning methods. *Sensors*, *21*(4), 1113.

[12] Wani, A., and Khaliq, R. (20210. SDN-based intrusion detection system for IoT using deep learning classifier (IDSIoT-SDL). *CAAI Transactions on Intelligence Technology*, *6*(3), 281-290.

[13] Fatani, A., Abd Elaziz, M., Dahou, A., Al-Qaness, M.A., & Lu, S. (2021). IoT intrusion detection system using deep learning and enhanced transient search optimization. *IEEE Access*, *9*, 123448-123464.

[14] Safaldin, M., Otair, M., & Abualigah, L. (2021). Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. *Journal of ambient intelligence and humanized computing*, *12*, 1559-1576.

[15] Ashiku, L., & Dagli, C. (2021). Network intrusion detection system using deep learning. Pro*cedia Computer Science*, *185*, 239-247.

[16] Rincy N, T., & Gupta, R. (2021). Design and development of an efficient network intrusion detection system using machine learning techniques. *Wireless Communications and Mobile Computing*, *2021*, 1-35.

[17] Imran, M., Haider, N., Shoaib, M., & Razzak, I. (2022). An intelligent and efficient network intrusion detection system using deep learning. *Computers and Electrical Engineering*, *99*, 107764.

[18] Kunang, Y.N., Nurmaini, S., Stiawan, D., & Suprapto, B.Y. (2021). Attack classification of an intrusion detection system using deep learning and hyperparameter optimization. *Journal of Information Security and Applications*, *58*, 102804.

[19] Sohi, S.M., Seifert, J.P., & Ganji, F. (2021). RNNIDS: Enhancing network intrusion detection systems through deep learning. *Computers & Security*, *102*, 102151.

[20] Gowdhaman, V., & Dhanapal, R. (2022). An intrusion detection system for wireless sensor networks using deep neural network. *Soft Computing*, *26*(23), 13059-13067.

[21] https://www.kaggle.com/datasets/hassan06/nslkdd

[22] Zhu, K., Wang, Z., Han, Q.L., & Wei, G. (2021). Distributed set-membership fusion filtering for nonlinear 2-D systems over sensor networks: An encoding–decoding scheme. *IEEE Transactions on Cybernetics*, *53*(1), 416-427.

[23] Cao, S., Qian, Q., Cao, Y., Li, W., Huang, W., & Liang, J. (2023). A Novel Meta-heuristic Algorithm for Numerical and Engineering Optimization Problems: Piranha Foraging Optimization Algorithm (PFOA). *IEEE Access*.

[24] Li, Y., Shi, T., Zhang, Y., & Ma, J. (2023). SPGAN-DA: Semantic-Preserved Generative Adversarial Network for Domain Adaptive Remote Sensing Image Semantic Segmentation. *IEEE Transactions on Geoscience and Remote Sensing*.

[25] Trojovský, P., Dehghani, M., & Hanuš, P. (2022). Siberian tiger optimization: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems. *Ieee Access*, *10*, 132396-132431.