Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

A prospective approach to detect advanced persistent threats: Utilizing hybrid optimization technique

Indra Kumari [a,b], Minho Lee [a,b,*]

[a] Department of Machine Learning Data Research, Korea Institute of Science and Technology Information (KISTI), Daejeon, 34141, Republic of Korea
[b] Department of Applied AI, University of Science and Technology (UST), Daejeon, 34113, Republic of Korea

ARTICLE INFO

Keywords:
Advanced persistent threats
Harris Hawk optimization
Sparrow search algorithm
Light GBM
Bi-LSTM classifiers

ABSTRACT

Advanced Persistent Threat (APT) attacks pose significant challenges for AI models in detecting and mitigating sophisticated and highly effective cyber threats. This research introduces a novel concept called Hybrid HHOSSA which is the grouping of Harris Hawk Optimization (HHO) and Sparrow Search Algorithm (SSA) characteristics for optimizing the feature selection and data balancing in the context of APT detection. In addition, the light GBM as well as the weighted average Bi-LSTM are optimized by the proposed hybrid HHOSSA optimization. The HHOSSA-based attribute selection is used to choose the most important attributes from the provided dataset in the early step of the quasi-identifier detection. The HHOSSA-SMOTE algorithm effectively balances the unbalanced data, such as the lateral movements and the data exfiltration in the DAPT 2020 database, which further improves the classifier performance. The light GBM and the Bi-LSTM classifier hyperparameters are well attuned and classified by the HHOSSA optimization for the precise classification of the attacks. The outcome of both the optimized light GBM and the Bi-LSTM classifier generates the final prediction of the attacks existing in the network. According to the research findings, the HHOSSA-hybrid classifier achieves high accuracy in detecting attacks, with an accuracy rate of 94.468 %, a sensitivity of 94.650 %, and a specificity of 95.230 % with a K-fold value of 10. Also, the HHOSSA-hybrid classifier achieves the highest AUC percentage of 97.032, highlighting its exceptional performance in detecting APT attacks.

1. Introduction

Advanced persistent threat (APT) attacks are a serious and harmful type of cyberattack that is frequently used to target business organizations and the key government [1]. APT is defined by three words: (1) Advanced: APT attackers are technologically advanced in various attack methods and attack tools expertise (2) Persistent: APT attackers are tenacious in their pursuit of the attack's goal. (3) Threat: The threat component of APT stems from the possibility of losing sensitive data or mission-critical components [2]. An APT is a highly advanced and focused cyberattack method used by knowledgeable adversaries with substantial resources and skills. APTs are distinguished by their determination, hiding, and prolonged nature. The fundamental goal of an APT is to acquire unauthorized access to sensitive data or systems and to keep it there for a long time while avoiding detection. Attackers find it challenging to mount threats

[*] Corresponding author. Department of Machine Learning Data Research, Korea Institute of Science and Technology Information (KISTI), Daejeon 34141, Republic of Korea.

E-mail address: cokeman@kisti.re.kr (M. Lee).

https://doi.org/10.1016/j.heliyon.2023.e21377
Received 25 May 2023; Received in revised form 10 October 2023; Accepted 20 October 2023
Available online 30 October 2023

against resources in the network traffic [3,4]. To detect APT in the network the authors test semi-supervised algorithms on the Dataset for Advanced Persistent Threats (DAPT 2020) dataset and demonstrate that they struggle to identify attack traffic at different stages of an advanced persistent threat [5,6]. As a result, any machine learning-based system must include defense measures against various attacks. Organizations have begun to take proactive measures to anticipate dangers. Detecting APTs is a difficult challenge for which no resolution has yet been identified due to the employment of a large variety of approaches and the difficulty of the attacks [7].

To overcome this challenge this research introduces a novel concept called hybrid HHOSSA that combines the characteristics of Harris Hawk Optimization [8] and Sparrow Search Algorithm [9] to optimize feature selection, data balancing, and model performance within the context of APT attack detection at an early stage. Harris Hawk Optimization, known for its sharp observation and analytical skills, represents the ability to carefully analyze and observe the dataset to identify the most significant attributes for APT attack detection. Similar to the keen vision and attention to detail of the Harris Hawk Optimization, the HHOSSA approach leverages this characteristic to select the most relevant attributes from the initial stage of the quasi-identifier detection. On the other hand, Sparrow Search Algorithms are known for their adaptability and agility. They possess the ability to navigate diverse environments and adapt to changing conditions. In the context of the HHOSSA approach, the characteristics of the Sparrow Search Algorithm are utilized to address the challenge of imbalanced data present in the DAPT 2020 attack dataset. The HHOSSA algorithm incorporates the adaptive and balancing capabilities of the Sparrow Search Algorithm, specifically through the implementation of the HHOSSA SMOTE algorithm. This algorithm effectively balances the imbalanced data and mitigates the lateral movements and data exfiltration, enabling a more comprehensive and accurate analysis of the dataset.

By combining the sharp observation and analytical skills of Harris Hawk Optimization with the adaptability and agility of the Sparrow Search Algorithm, the HHOSSA approach aims to optimize the entire APT attack detection process. It not only focuses on selecting the most relevant attributes but also addresses the challenges of imbalanced data ultimately enhancing the performance of the Light GBM and Bi-LSTM classifiers used for APT attack detection. Light GBM is made to be much more computationally efficient. For feature discretization, it employs a histogram-based method, which utilizes less memory and speeds up training. This facilitates faster experimentation and model iteration and makes it appropriate for handling large-scale datasets. Due to its bidirectional nature, Bi-LSTM models by default gather contextual data from both past and future inputs. Hence these two classifiers are integrated to get fast training with less memory space. The HHOSSA optimization is proposed by integrating the behavior combination of two birds which helps the classifier by providing an astonishing result of the high convergence rate and less computational time.

Hence, in this research, HHOSSA optimization is proposed for attribute selection involved in the data and data balancing using the HHOSSA SMOTE algorithm. The significant parameters present in the light GBM as well as the Bi-LSTM classifier are optimized and well-trained by the proposed HHOSSA optimization. The time domain-based statistical features and the relevant data attributes are selected to enhance the performance of the HHOSSA-hybrid classifier in advanced persistent threat attack detection. The major contribution to this research is,

a) **HHOSSA-based attribute selection:** The most significant attributes present in the provided dataset are selected by the proposed HHOSSA-based attribute selection from the initial stage of the quasi-identifier detection.
b) **HHOSSA-SMOTE and Hybrid Classifier:** The imbalanced data, including lateral movements and the data exfiltration present in the DAPT 2020 database are well balanced by the proposed HHOSSA SMOTE algorithm. Light GBM is made to be much more computationally efficient. For feature discretization, it employs a histogram-based method, which utilizes less memory and speeds up training. This facilitates faster experimentation and model iteration and makes it appropriate for handling large-scale datasets. Due to its bidirectional nature, Bi-LSTM models by default gather contextual data from both past and future inputs. Hence these two classifiers are integrated to get fast training with less memory space.
c) **HHOSSA Optimization:** The optimization is a behavioral combination of Harris Optimization and Sparrow Search Algorithm where the Harris Hawk is the only predator bird that hunts in the group that has the advantage of best hunting capacity and unexpected attacks these behaviors are integrated with the best global position of Sparrow Search Algorithm that helps to provide the best optimal solution and balances the data and finds the attack accurately.

The remainder of the paper is arranged as follows: the review of the existing techniques for attack detection with the evolved challenges is described in section 2. Section 3 explained the proposed method for attack detection using HHOSSA optimization for the Light GBM and Bi-LSTM classifier. Section 4 revealed the results and discussion of the existing and proposed method. Finally, the paper is concluded in section 5.

2. Related work

The detection of Advanced Persistent Threats (APTs) has become a significant concern for organizations and researchers in the field of cybersecurity. APTs are long-term and targeted cyber-attacks that aim to gain covert access and control over a network.

Chowdhary et al. [10] address the need for a benchmark dataset specifically designed for modeling and detecting Advanced Persistent Threats (APT). The authors highlight the limitations of generic intrusion datasets in capturing the complexity and sophistication of APT attacks. To overcome these limitations, the authors propose the DAPT 2020 dataset, which consists of attacks that are part of APTs. The dataset includes traffic from both the public-to-private interface and the internal network, reflecting the diverse attack vectors employed in APT scenarios. By benchmarking the DAPT 2020 dataset on semi-supervised models, the authors demonstrate the challenges of detecting APT attacks, particularly in the presence of severe class imbalance. The DAPT 2020 dataset provides a valuable resource for researchers and organizations working on APT detection. Sailik et al. [11] focus on APT attacks, and the inclusion of diverse attack vectors makes it more representative of real-world scenarios. The dataset's utilization of semi-supervised models highlights the difficulties in accurately detecting APT attacks, further emphasizing the need for novel approaches and techniques in this domain. Overall, the DAPT 2020 dataset contributes to advancing the field of APT detection by addressing the limitations of existing datasets and providing a foundation for the development and evaluation of more effective machine-learning models tailored to detect APT attacks.

Wang et al. [12] the authors address the challenge of identifying the different stages of Advanced Persistent Threat (APT) attacks. APT attacks are covert and targeted attacks that pose serious security risks to vital information systems. Traditional intrusion detection systems struggle to capture the behavioral features of APT attacks, making it difficult to detect and mitigate them effectively. To overcome this challenge, the authors propose APTSID, an ensemble learning method based on network traffic analysis. The APTSID method utilizes an anomaly detection-attack stage detection pipeline to achieve multi-stage identification of APT attacks. By accurately identifying the attack stage, security engineers can understand the implemented damage and devise precise defense strategies. Experimental results demonstrate that the APTSID method outperforms traditional machine learning algorithms in terms of detection performance. The proposed approach offers promising prospects for enhancing APT attack detection and establishing a more robust defense against these sophisticated threats.

Alrehaili et al. [13] elaborate that the authors propose an efficient and flexible deep-learning model for detecting indications of APT attacks by analyzing network traffic. They employ a hybrid approach that combines a Stacked Autoencoder with Long Short-Term Memory (SAE-LSTM) and Convolutional Neural Networks with Long Short-Term Memory Network (CNN-LSTM). Using the reliable dataset 'DAPT2020,' which covers all APT stages, the experimental results demonstrate that the hybrid deep learning approach outperforms individual models in detecting malicious behavior at each APT stage. Serkan et al. [14] and SAVAŞ et al. [15] proposed this research highlights the efficacy of machine learning techniques in detecting phishing attacks. By leveraging various machine learning algorithms and utilizing email features, the proposed system demonstrates robustness and accuracy in distinguishing malicious phishing emails. The study emphasizes the importance of machine learning in combating cybersecurity threats beyond APTs, specifically in the context of phishing attacks.

Meng et al. [16] proposed the research emphasizes the significance of cybersecurity governance and management in addressing cyber threats. They emphasize the need for careful handling of attacks and the importance of making informed decisions based on data. By conducting systematic literature reviews and surveys, these studies contribute to the understanding of best practices, frameworks, and approaches that can enhance cybersecurity governance and risk management. The findings from these articles underscore the importance of a proactive and well-structured governance framework to effectively protect against cyber threats. Dijk et al. [17] authors explore the limitations of signature-based IDSs and the need for anomaly-based detection in APT scenarios. They highlight that

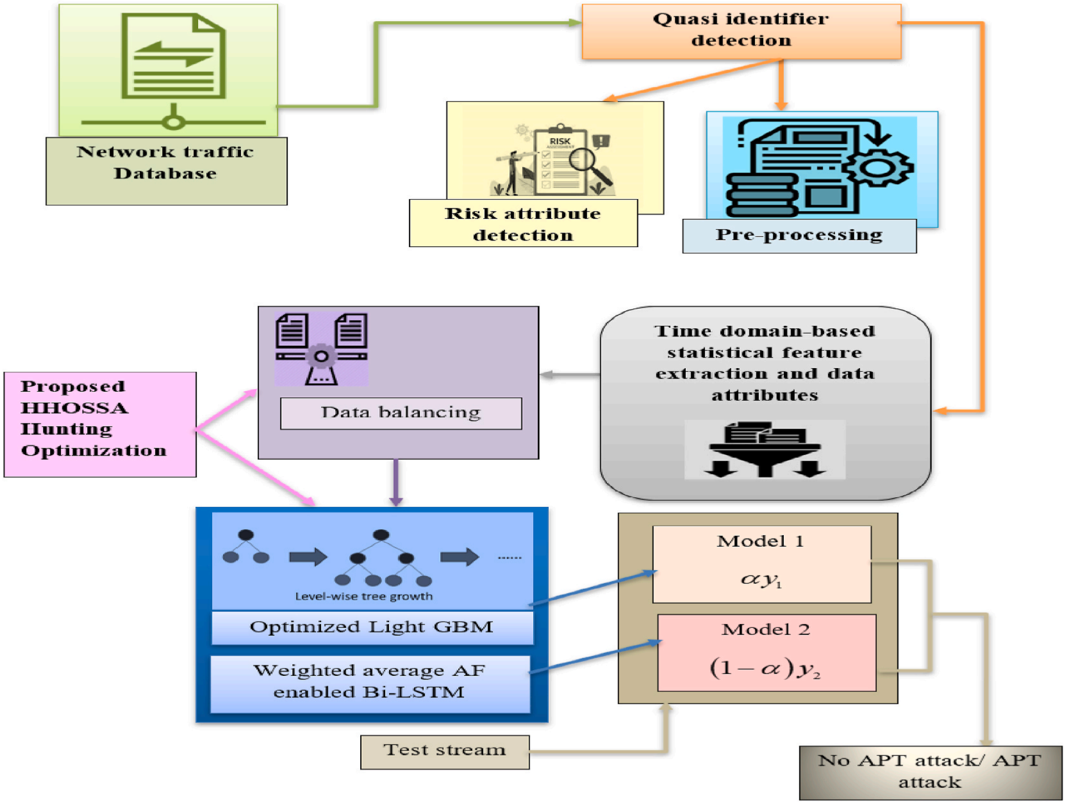

Network traffic Database
Quasi identifier detection
RISK
Risk attribute detection
Pre-processing
Time domain-based statistical feature extraction and data attributes
Proposed HHOSSA Hunting Optimization
Data balancing
Level-wise tree growth
Optimized Light GBM
Weighted average AF enabled Bi-LSTM
Model 1
αy_1
Model 2
$(1-\alpha)y_2$
Test stream
No APT attack/ APT attack

Fig. 1. Advanced persistent threat attack detection.

signature-based IDSs primarily protect against known attacks, leaving organizations vulnerable to unknown malware and zero-day vulnerabilities. APT attacks, characterized by their long-term nature and state-sponsored origins, require detection mechanisms that can differentiate between benign and unseen attack traffic. The paper presents the analysis of different APT stages, including reconnaissance, foothold establishment, lateral movement, and data exfiltration. Data exfiltration, the act of stealing non-public and high-value data from a network, is the focus of the proposed AI-based detection method. By leveraging DPI and AI techniques, significant improvements in the detection of anomalies in the data exfiltration APT stage are demonstrated.

Overall, the related work demonstrates the growing interest in addressing the challenges posed by APT attacks through the use of advanced techniques such as deep learning and deep packet inspection. These approaches aim to overcome the limitations of signature-based detection, detect complex and targeted attacks, and differentiate between benign and malicious network traffic. The proposed methods show promising results in terms of accuracy and detection performance on specialized APT datasets, paving the way for further advancements in APT detection and mitigation.

3. Proposed methodology

Datasets are critical for developing neural network models that can identify advanced and complicated risks like APT. APT datasets, however, that may be utilized for modeling and detecting APT attacks are not yet available. APT attacks are included in the dataset DAPT 2020 which are (1) difficult to differentiate from typical traffic streams and (2) include traffic on both the private networking and the public-to-private interfaces. Data movement through a computer network is referred to as network traffic. Network traffic, also known as data traffic, is split up into packets of data and transferred over a system before even being pieced back together by the computer or device that is receiving it. One of the most negative consequences of a network traffic issue is data loss and the packet loss results from data traffic when an Internet route is overloaded. There is no more data that can be transmitted over the entire Internet route and the data packets didn't even reach the receiving system as a consequence. Imbalanced data in the DAPT 2020 dataset should be improved which has a low sample size in the lateral movements and the data exfiltration.

The proposed schematic representation for the four different types of attack detection is illustrated in Fig. 1. The network traffic database is provided as input to determine the attacks and then the Quasi-identifiers are identified based on the availability of risk attributes in the database. After determining the risk attributes in the database, the attributes are preprocessed to recognize the missing values, and also the significant features are designated. From the selected features, the statistical features are extracted depending on the time domain-based technique to gather the required features and the data attributes are also extracted. The extracted statistical features and the data attributes have the issue of data imbalance which is resolved by the proposed HHOSSA optimization. The optimized light GBM and the weighted average AF-enabled Bi-LSTM classifier are well-tuned by the proposed HHOSSA optimization which is separately trained and modeled. The output of the two trained models is αy_1 and $(1-\alpha)y_2$ which is compared with the test stream to identify the attackers in the database, where the y_1 and y_2 are the output of the Light GBM as well as the Bi-LSTM classifier, α be the random variable generated by the trial and error method.

3.1. DAPT 2020 dataset

The DAPT 2020 dataset gathers the five publicly available data during the period of Monday to Friday with five various types of attacks. The data is provided in.csv (comma-separated values) file format, which is commonly used for storing tabular data. The DAPT 2020 datasets focus on key characteristics of APT attacks, such as persistence, slow movement, and low visibility. The attacks were carried out on Tuesday, Wednesday, Thursday, and Friday, while normal traffic was generated throughout the day on Monday. The files are the dimensions of 8728×85, 29242×85, 17487×85, 9685×85, and 7361×85. The DAPT 2020 dataset has 45 float, 34 int, and 6 object features available. From the available features, the irrelevant features are removed during the preprocessing stage of the Quasi-identifier detection, that is the flow ID as well as the time stamp. The labels, such as the activity and the stage are removed for enhancing the performance of DAPT attack detection and now the attributes are reduced from 85 to 81. The time domain-based statistical features and the data attributes are integrated into the existing 81 attributes, now the attributes are increased to 85. Then, the attained feature vector with the dimension of 72503×85 and were the label size as 72503×1. The benign, data exfiltration, established foothold, lateral movement, and reconnaissance attack are involved in the provided DAPT 2020 database. The data exfiltration and the lateral movement have imbalanced data, in which the data are kept balanced using the proposed HHOSSA optimization. Let C be the input streaming traffic data with $''n''$ entries each consisting of $''m''$ attributes and is formulated as

$$C^t = \left\{ d_g^t[i]; \quad 1 \le g \le n; \quad 1 \le i \le m \right. \tag{1}$$

where the time series data are represented as $d_g^t[i]$ in equation (1).

3.2. Quasi-attribute detection using the proposed risk attribute ranking approach

The risk attributes are detected using the proposed risk attribute ranking approach in the quasi-identifier detection, initially, the identified attributes are preprocessed to remove the irrelevant attributes present in the DAPT database [18], which are described as follows,

3.2.1. Data preprocessing

At first, 85 attributes are present in the DAPT 2020 database, after preprocessing the provided data in the quasi-identifier detection, the 85 attributes are reduced to 81 by eliminating the two irrelevant attributes, such as flow-id and stamp, in addition to eliminating the two labels as activity and stage. Irrelevant attributes in the raw data are modified in an understandable way to enhance data quality. The initial step of pre-processing involves filling in any missing numbers or values, correcting dataset discrepancies, and standardizing the data. The transformation of sample variables yields numerical and ordinal values, and the removal of unbalanced data occurs during pre-processing before further processing.

3.2.2. Proposed risk attribute ranking approach

To identify the similarity between two different data attributes, the risk factor *r* is proposed in this approach, and is formulated as,

$$r = \mathrm{Cos}\left(d_g[i,k]\right) + Eu\left(d_g[i,k]\right) + BA\left(d_g[i,k]\right) \tag{2}$$

where the attribute *"i"* is declared as a significant attribute, if the risk factor is higher than the threshold value or else the attribute is declared as an insignificant or redundant attribute, cosine similarity is denoted as Cos [19], Euclidean [20] and Bhattacharya distance [21] are represented as *Eu* and *BA*. The k^{th} attribute in the dataset is represented as *k* in equation (2). The most significant steps involved in the proposed risk attribute ranking approach are as follows:

Step 1. *Attribute centroid initialization:* The attribute centroids are initialized as,

$$d = \{d_1, d_2, ...d_i, ..., d_v\} \tag{3}$$

where, the attributes are randomly initialized for the detection of the Quasi attribute in the dataset, and the total clusters or the significant attributes are denoted as *v* in equation (3).

Step 2. *Estimate the similarity (Fitness Function):* Call *"r"* the rank of the attribute to evaluate the similarity between the i^{th} as well as the k^{th} attribute.

Step 3. *Ranking the attributes:* The attributes are ranked depending on the value of the estimated risk factor.

Step 4. Declare the quasi-attributes of the dataset and terminate. Finally, the selected attributes are represented as,

$$C^t = \left\{ d_g^{t,S}[i]; \quad 1 \le S \le Q; \quad 1 \le g \le n \right. \tag{4}$$

where the quasi attributes are represented as *S* in equation (4). The attribute selection procedure is performed optimally using the proposed optimization discussed in section 4.6.

3.3. Formation of attribute vector (time domain-based statistical feature extraction)

The features from the selected attributes are extracted by utilizing the time domain-based statistical feature extraction for the formation of the attributes vector, and the features are extracted by assigning a particular time in the dataset.

3.3.1. Mean

The mean value is the sum of the significant attributes to the total number of quasi-attributes present in the dataset.

$$\mu_g^t = \frac{1}{Q}\sum_{i=1}^{Q} d_g^{t,S}[i] \tag{5}$$

where the quasi attributes are denoted as *S*, the significant attributes are denoted as *i*, and the total number of quasi attributes present in the data are represented as *Q* at a time instance of *t* in equation (5).

3.3.2. Kurtosis

Kurtosis K is estimated by the ratio of mean value to the standard deviation considering the time series data $d_g^{t,S}[i]$ in equation (6) as,

$$\mathrm{K}_g^t = \frac{\frac{1}{Q}\sum_{i=1}^{Q} d_g^{t,S}[i]}{\sigma} \tag{6}$$

where σ represents the standard deviation in equation (7) which is formulated as,

$$\sigma = \sqrt{\frac{1}{Q}\sum_{i=1}^{Q}\left(d_g^{t,S}[i] - \mu_g^t\right)} \tag{7}$$

3.3.3. Skewness

The skewness is used to measure the lack of symmetry in the provided data, which depends on the positive and negative signs in the distribution of signals which is formulated in equation (8) as,

$$u_g^t = \frac{1}{Q\sigma^3}\sum_{i=1}^{Q}\left(d_g^{t,S}[i] - \mu_g^t\right)^3 \tag{8}$$

3.3.4. Variance

The square of the standard deviation is represented as the variance, and is formulated in equation (9) as,

$$v_g^t = \frac{1}{Q}\sum_{i=1}^{Q}\left(d_g^{t,S}[i] - \mu_g^t\right) \tag{9}$$

The attribute vector depending on the time domain-based statistical feature from the dataset is formulated in equation (10) as,

$$\mathrm{A}^g = \left\{\mu_g^t, \mathrm{K}_g^t, u_g^t, v_g^t, C^t\right\};\ i \le Q;\ g \le n \tag{10}$$

3.4. Data balancing using HHOSSA-SMOTE algorithm

In the DAPT 2020 dataset, the benign attack, data exfiltration, established foothold, lateral movement, and reconnaissance attacks are involved with the adequate amount of data as 51848, 9, 8600, 137, and 11909. The data exfiltration attack data as well as the lateral movement data are not balanced with less amount of data. To balance the imbalanced data, the proposed HHOSSA SMOTE is utilized to attain better classification accuracy in attack detection from the DAPT 2020 dataset. The HHOSSA SMOTE method is initially trained on the unbalanced data for a specific sample that is present in both the data characteristics and the temporal domain-based statistical features. The fitness function is then assessed for each sample once optimal sampling has ended and there are no longer any training data available. After the termination of optimum sampling, training data are absent, and then the fitness function is evaluated for each sample. Depending on the fitness function, the data to be trained and the attain the balanced data for further processing. According to the fitness function, the training data, and the goal of obtaining balanced data for further analysis have been done.

3.5. APT attack detection

The light GBM classifier is easily implemented but the tuning of hyperparameters is quite difficult, some of the commonly used parameters are control parameters, core parameters, metrics, and the IO parameter. These parameters are tuned by the HHOSSA optimization for achieving better operation in detecting the attack as normal or abnormal. The light GBM tends to handle a large amount of data with higher accuracy by minimizing the loss function using the decision tree structure of the optimized light GBM and attaining a significant output. The performance of detecting the attacks is additionally enhanced by integrating the optimized Bi-LSTM classifier with the accurate output.

3.5.1. Optimized light gradient boosting machine for attack detection

To minimize the loss function $G(l,p(x))$, light GBM [22] tends to identify the approximation $\widehat{p}(x)$ to a particular function $p^*(x)$ in equation (11) as follows,

$$\widehat{p} = \arg\min_{p}\mathrm{I}_{l,x}G(l,p(x)) \tag{11}$$

To estimate the final model, the light GBM assimilates the various w regression trees $\sum_{z=1}^{w}p_z(\mathrm{P})$, which are formulated in equation (12) as,

$$p_w(\mathrm{P}) = \sum_{z=1}^{w}p_z(\mathrm{P}) \tag{12}$$

The regression trees are described as, $\mathrm{O}_{e(x)}, e \in \{1,2,\ldots.o\}$, where the presence of leaves are denoted as o, the rules that arise for the decision tree are denoted as e, and the leaf nodes sample weight is represented as a vector O. Thus, the light GBM is trained with the step size of z as follows,

$$\Gamma_z = \sum_{b=1}^{q}G\left(l_b, J_{z-1}(x_b) + p_z(p_b)\right) \tag{13}$$

The constant in equation (13) can be eliminated by introducing the loss function gradient statistics, and is renovated as follows,

$$\Gamma_z \cong \sum_{b=1}^{q}\left(c_b p_z(x_b) + \frac{1}{2}N_b p_z^2(x_b)\right) \tag{14}$$

where the loss function gradient statistics in the first and second order is represented as c_i and N_b, the leaf o sample set is denoted as X_o, and then equation (14) can be described in equation (15) as follows,

$$\Gamma_z = \sum_{o=1}^{o}\left(\left(\sum_{b\in \mathrm{X}_o} c_b\right)\mathrm{O}_o + \frac{1}{2}\left(\sum_{b\in \mathrm{X}_o} N_b + \lambda\right)\mathrm{O}_o^2\right) \tag{15}$$

The maximum value of Γ_z can be determined by considering the loss function gradient statistics in the first and second order in equation (16) for a particular decision tree $e(x)$ with the best weight score of an individual node O_o^*.

$$\mathrm{O}_o^* = -\frac{\sum_{b\in \mathrm{X}_o} c_b}{\sum_{b\in \mathrm{X}_o} N_b + \lambda} \tag{16}$$

$$\Gamma_w^* = -\frac{1}{2}\sum_{o=1}^{o}\frac{\left(\sum_{b\in \mathrm{X}_o} c_b\right)}{\sum_{b\in \mathrm{X}_o} N_b + \lambda} \tag{17}$$

The quality of the decision tree structure e is measured by utilizing the scoring function Γ_w^* in equation (17), and the objective function attained from the light GBM is y_1 after integrating the splitting function in equation (18) as,

$$y_1 = \frac{1}{2}\left(\frac{\left(\sum_{b\in \mathrm{X}_G} c_b\right)^2}{\sum_{b\in \mathrm{X}_G} N_b + \lambda} + \frac{\left(\sum_{b\in \mathrm{X}_Z} c_b\right)^2}{\sum_{b\in \mathrm{X}_Z} N_b + \lambda} - \frac{\left(\sum_{b\in \mathrm{X}} c_b\right)^2}{\left(\sum_{b\in \mathrm{X}} N_b + \lambda\right)}\right) \tag{18}$$

The samples involved in the right and left branches of the decision tree are represented as X_G and X_Z. Light GBM processes vast amounts of features and data efficiently since it grows the tree vertically rather than horizontally, as do other methods. In general, hyper-parameters would have a considerable impact on predicting accuracy. Therefore, establishing the range as well as the number of fluctuations of its hyper-parameter before employing Light GBM.

3.5.2. *Weighted average AF enabled Bi-LSTM for attack detection*

When processing the provided input data by the LSTM network, the data processing occurs sequentially, and only recovers the data available above the appropriate sample depth, while the data present below the sample depth is lost. The existing LSTM classifier need not encode the data from back to front thus, processing the data in reverse order is also significant. The Bi-LSTM [23] enhances the LSTM network by integrating the forward as well as the reverse LSTM to create a two-way RNN model. Initially, the input data is fed forward in unidirectional which is the forward transmission, and after the data is processed in the reverse order that is the backward

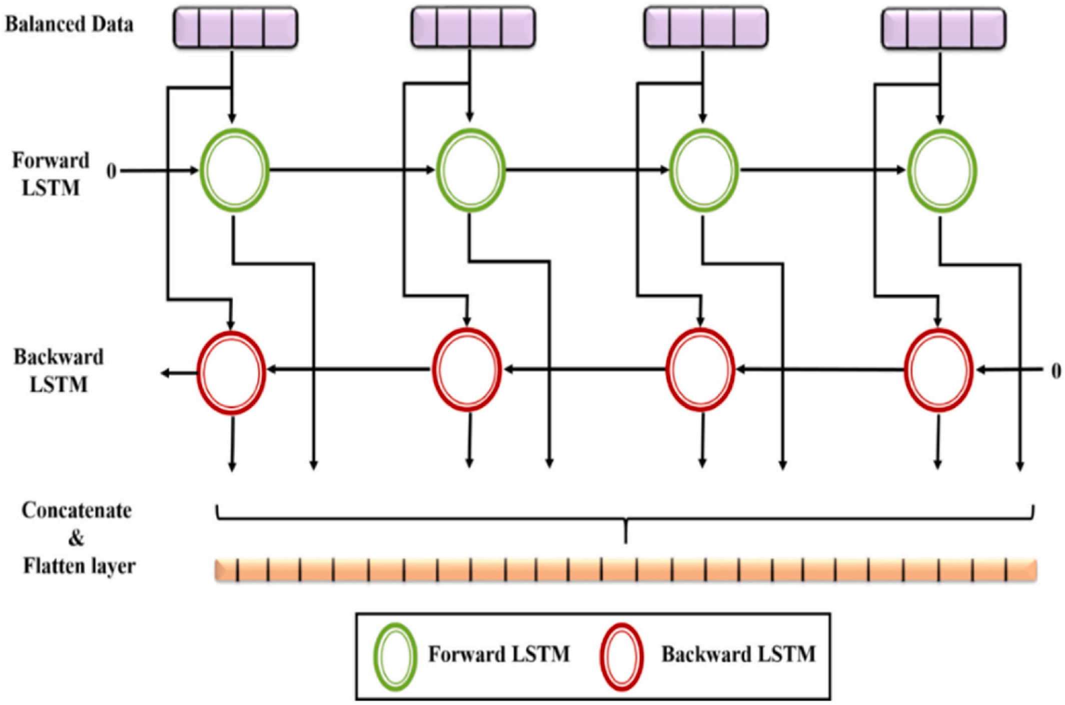


Fig. 2. Bi-LSTM architecture for attack detection.

transmission in vice versa to enhance the performance of the classifier. To attain better performance, and also to well-tune the hyperparameters available in the Bi-LSTM classifier revealed in Fig. 2 which is optimized by the proposed HHOSSA optimization. This optimization assists to maintain the average weight of the Bi-LSTM network and the final output of the Bi-LSTM network with both the backward as well as forward directions are described as y_2.

The final output detection of both the optimized light GBM and the weighted average AF-enabled Bi-LSTM classifier is described in equation (19) as,

$$\Upsilon = \alpha_1 * y_1 + (1 - \alpha_1) y_2 \tag{19}$$

where, the random factor is represented as α_1, which is within the range of 0–1.

Fig. 2 showcases the backward layers, which play a crucial role in the model. These backward layers operate in conjunction with the forward layers to capture contextual information from both past and future time steps. By processing the input sequence in reverse order, the backward layers enable the model to capture dependencies and patterns that may not be captured by the forward layers alone. This bidirectional approach enhances the model's ability to understand and detect attack patterns in the given data.

3.6. Proposed HHOSSA optimization for APT attack detection

The proposed HHOSSA optimization is the grouping of Harris Hawk Optimization and Sparrow Search Algorithm characteristics for optimizing the feature selection and data balancing. In addition, the light GBM as well as the weighted average AF-enabled Bi-LSTM are optimized by the proposed HHOSSA optimization. The Harris Hawk Optimization has assisted with one or more factors individually or all of them combined for improving performance. First, HHOSSA optimization is a technique that benefits from the time-varying elements and has a very positive result on effectiveness and improves the harmony of the sturdiness of the exploring cores across the iterations when the escaping energetic component has such a dynamic randomized time-varying attribute. At almost the same time, this property enables HHOSSA optimization to convert the aforementioned phases completely. Second, HHOSSA optimization features a multiphase (flexible) exploring stage (global search) which also takes into account the center of mass or the average location of Harris Hawks, this feature can make the process more effective and innovative during the initial iterations. The third crucial characteristic is a variety of levy-triggered patterns with different bouncing configurations throughout the exploitation stage. This bouncing potential has improved the local search's area and depth in practically all HHOSSA optimization variations as well as in its initial form. The fourth feature results from the incremental selection strategy used during the random walk enhanced search propagation. This skill enables exploration agents (Harris Hawk Optimizations) to increase the range of their space travel while only choosing the best move. In the swindling stage, the HHOSSA optimization multiphase design makes it simpler to access a wider range of intermittent, brief patterns. As a result, if one enclosing approach fails, another can be used, and ultimately, the finest is preserved for development in the following iteration. The next characteristic is a result of the skillfully created randomized bouncing velocity, which has also helped to further integrate local, and global exploration as well as local optima avoidance.

Like every randomized population-based optimization, the HHOSSA optimization has some restrictions for its initial potential in addition to all of its benefits. The first is that a suitable adjustment will be required to speed up the exploring stage of the algorithm when the population of the HHOSSA optimization is constrained to local optima for a challenging task. As a result, HHOSSA optimization cannot always be free from local optima and sometimes displays the results or exhibits a tendency for convergence that is premature. Another drawback is that HHOSSA optimization occasionally struggles to maintain the appropriate balance between the centers, identify the true instances, or make the shift smoothly, particularly when working with extremely complex attribute spaces. These limitations are overcome by grouping the Sparrow Search Algorithm's characteristics with the Harris Hawk Optimization which enhances the performance of better DAPT attack detection.

3.6.1. Inspiration

The Harris Hawk Optimization behavior during exploration, unexpected attack, and hunting strategy served as an inspiration for the construction of the HHOSSA optimization as well as the exploring and swindling trends of the traditional form. The three significant phases involved in the hunting behavior of Harris Hawk Optimization are the exploration, changeover from exploration to the swindling stage, and the swindling. While other birds sometimes hunt alone, the Harris Hawk Optimization is unique in that it hunts together in tolerant bands. The intellectual ability of Harris Hawk Optimizations, which enables them simple to learn and has produced such a well-liked bird, has been related to their interactive nature. This species inhabits reasonably steady populations. In Harris Hawk Optimization, there is a predominance hierarchy in which the adult female is the dominating bird, preceded by the adult male and then the offspring from earlier years. Normally, 2 to 7 birds form a group in which the birds work together during both the hunting and nesting processes. This species of bird of predator is the only one known to regularly hunt in groups. Harris Hawk Optimizations hunt in communal groups of two to six, unlike the majority of birds, which are independent and only congregate during migration and hatching. This is thought to be adaptable to the absence of prey in their native desert environment. One method of hunting involves a small group flying ahead and scouting, followed by another member of the group flying ahead and scouting, and so on until the target is caught and distributed. In another, the prey is surrounded by all the Harris Hawks, and one of them chases it out. Because most birds do not even spend lots of time on the land, Harris Hawk Optimizations frequently pursue prey on foot and are particularly swift on the ground, their extended, yellow legs are designed for this.

3.6.2. *Mathematical modeling of the HHOSSA optimization*

In the HHOSSA optimization, the major steps involved in attacking the targeted prey are the exploring stage, shifting from exploring to swindling, and the swindling stage with the four tactics soft encircle, hard encircle, soft encircle with advanced rapid dives, hard encircle with advanced rapid dives.

3.6.2.1. *Exploring stage.*

In Fig. 3 the exploring stage explains the Harris Hawk location while exploring their targeted prey which is based on the two different tactics. Depending on the location of the real members, the exploring stage initial tactics explain how the Zen folio identifies the prey which is described in equation (21). Depending on the presence of the Harris Hawk in the branch of a random tree M_{rand}, the second tactic in the exploring stage explains how the Harris Hawk identifies the targeted prey which is also described in equation (21). The location of the real members is initialized in equation (20) as,

$$\mathrm{M}_j, j = 1, 2, 3, 4, \ldots.., \mathrm{H} \tag{20}$$

where the total number of Harris Hawk in the proposed HHOSSA optimization is denoted as H.

$$\mathrm{M}_j^{\mathrm{T}+1} = \begin{cases} \mathrm{M}^{\mathrm{T}}_{rand} - R_1 \left|\mathrm{M}^{\mathrm{T}}_{rand} - 2R_2\mathrm{M}^{\mathrm{T}}\right|, h \geq 0.5 \\ \left(\mathrm{M}^{\mathrm{T}}_{prey} - \mathrm{M}^{\mathrm{T}}_{a} - D\right), h < 0.5 \end{cases} \tag{21}$$

where the renovated location of Harris Hawk is represented as $\mathrm{M}_j^{\mathrm{T}+1}$ in the next iteration T, the present location of Harris Hawk is represented as $\mathrm{M}^{\mathrm{T}}_{rand}$, the random members available in the set of (0,1) is denoted as $R_1, R_2, R_3, R_4,$ and h, the location of the targeted prey is denoted as $\mathrm{M}^{\mathrm{T}}_{prey}$, and the average location of all the Harris Hawk is denoted as $\mathrm{M}^{\mathrm{T}}_{a}$ which is described in equation (22) as follows,

$$\mathrm{M}^{\mathrm{T}}_{a} = \frac{\sum_j^{\mathrm{H}} \mathrm{M}_j^{\mathrm{T}}}{\mathrm{H}} \tag{22}$$

where the variables of upper and lower bounds differences are represented in equation (23) as,

$$D = R_3(L + R_4(U - L)) \tag{23}$$

The renovated location of the Harris Hawk in the exploring stage is renovated by integrating the present global best location of the Sparrow Search from the standardized equation of (33) in Ref. [24].

Case 1. $h \geq 0.5$

From equation (21), the renovated location of the HHO for $h \geq 0.5$ is represented as follows,

$$\mathrm{M}_j^{\mathrm{T}+1} = \mathrm{M}^{\mathrm{T}}_{rand} - R_1 \left|\mathrm{M}^{\mathrm{T}}_{rand} - 2R_2\mathrm{M}^{\mathrm{T}}\right| \tag{24}$$

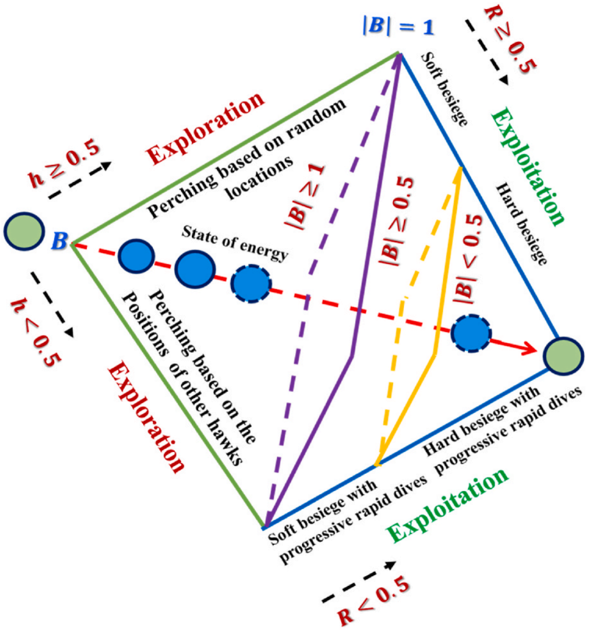

|B| = 1
R ≥ 0.5
Soft besiege
Exploitation
Hard besiege
Exploration
Perching based on random locations
h ≥ 0.5
B
State of energy
|B| ≥ 1
|B| ≥ 0.5
|B| < 0.5
h < 0.5
Perching based on the Positions of other hawks
Exploration
Soft besiege with progressive rapid dives
Hard besiege with progressive rapid dives
Exploitation
R < 0.5

Fig. 3. Different phases of HHO (Heidari et al., 2019).

The present global position of SSA is represented as,

$$M_f^{T+1} = M_{best}^{T} + \beta|M^{T} - M_{best}| \tag{25}$$

where the control parameter for the step size is described as β, and the present global position of Sparrow Search is represented as M_{best}.

The standardized form of an equation for enhancing the performance of HHOSSA optimization is formulated in equation (26) as,

$$M^{T+1} = 0.5\ M_j^{T+!} + M_f^{T+1} \tag{26}$$

$$M^{T+1} = 0.5\left[M_{rand}^{T} - R_1\left|M_{rand}^{T} - 2R_2M^{T}\right| + M_{best}^{T} + \beta|M^{T} - M_{best}|\right] \tag{27}$$

Let us as assume that || is the appropriate value in equation (27), and equation (28) is rewritten as follows,

$$M^{T+1} = 0.5\left[M_{rand}^{T} + R_1\left|M_{rand}^{T} - 2R_2M^{T}\right| + M_{best}^{T} + \beta|M^{T} - M_{best}|\right] \tag{28}$$

$$M^{T+1} = 0.5\left[M_{rand}^{T}(1 + R_1) - 2R_1R_2M^{T} + M_{best}^{T} + \beta M^{T}\right] \tag{29}$$

$$M^{T+1} = 0.5\left[M_{rand}^{T}(1 + R_1) + M^{T}(\beta - 2R_1R_2) + M_{best}^{T}(1 - \beta)\right] \quad h \geq 0.5 \tag{30}$$

thus, the renovated position of the Harris Hawk is improved by integrating the Sparrow Search global best location during the exploration stage in equation (29) which further improves the tactic in the detection of prey formulated in equation (30).

3.6.2.2. Shifting from exploring to the swindling stage. In this stage, the shifting characteristics of Harris Hawk from the exploring to the swindling stage are described which is based on the energy of the targeted prey during their escaping behavior in equation (31) as follows.

$$B = 2B_0\left(1 - \frac{T}{t}\right) \tag{31}$$

Initially, the energy of the targeted prey is denoted as B_0 which randomly varies in the range of -1 and 1. The energy of the targeted prey by the Harris Hawk is considered maximum if the energy is increased from 0 to 1. The energy of the targeted prey by the Harris Hawk is considered as a minimum if the energy decreases to -1 from 0. Depending on the value of B, the exploring stage, as well as the swindling stage, is considered, if the value of $|B| \geq 1$, then the exploring stage will be uncompleted, and if the value of $|B| < 1$, then the swindling stage is occurring.

3.6.2.3. Swindling stage. In the swindling stage, the significant hunting tactics of the Harris Hawk and the escaping behavior of the targeted prey are the two major elements. The Harris Hawks' unexpected pounce is designed in this stage with the four significant tactics soft encircle, hard encircle, soft and hard encircle with advanced rapid dives. The following tactics are involved in the swindling stage of the proposed HHOSSA optimization,

3.6.2.4. Tactic 1: soft encircle. At the condition of energy as $|B| \geq 0.5$ and the random members as $R \geq 0.5$, there is the occurrence of soft encircle behavior of Harris Hawk. This indicates that the targeted prey is unable to effectively move since the Harris Hawks deplete their energy during the process of soft encircling. Then equation (32) formulates the model for the soft encircling behavior of the Harris Hawk,

$$M_j^{T+1} = \Delta M^{T} - B\left|FM_{prey}^{T} - M^{T}\right| \tag{32}$$

$$\Delta M^{T} = M_{prey}^{T} - M^{T} \tag{33}$$

$$M_j^{T+1} = M_{prey}^{T} - M^{T} - B\left|FM_{prey}^{T} - M^{T}\right| \tag{34}$$

$$M_j^{T+1} = M_{prey}^{T}(1 - BF) - M^{T}(1 + B) \tag{35}$$

where the rabbit is considered as the target here and the variations between the rabbit position vector as well as the present location in the iteration T are denoted as ΔM^{T}. The escaping tactics of the targeted prey are represented as $F = 2(1 - R_5)$ which varied randomly depending on every iteration formulated in equation (34) and equation (35). The random member is denoted as R_5 within the range of 0 and 1.

In the swindling stage, the soft encircle tactic that is equation (14) is combined with the Sparrow Search global best present location in equation (25) with the standardized form in equation (36) as follows,

$$M^{T+1} = 0.5M_j^{T+1} + 0.5M_f^{T+1} \tag{36}$$

$$M^{T+1} = 0.5\left\{M^{T}_{prey}(1 - BF) - M^{T}(1 + B) + M^{T}_{best} + \beta(M^{T} - M_{best})\right\} \tag{37}$$

$$M^{T+1} = 0.5\left\{M^{T}_{prey}(1 - BF) - M^{T}(1 + B) + M^{T}_{best} + \beta M^{T} - \beta M_{best}\right\} \tag{38}$$

$$M^{T+1} = 0.5\left\{M^{T}_{prey}(1 - BF) - M^{T}(1 + B + \beta) + M^{T}_{best}(1 - \beta)\right\} \tag{39}$$

thus, equation (37), equation (38) and equation (39) are modified by the Sparrow Search behavior, the hunting behavior of the Harris Hawk is improved in the initial tactic of the swindling stage for catching the targeted prey which reduces the consumption time of tuning the classifier parameters.

3.6.2.5. Tactic 2: hard encircle. At the condition of energy as $|B| < 0.5$ and the random members as $R \geq 0.5$, there is the occurrence of hard encircle behavior of Harris Hawk, which indicates that the targeted prey is tired out and unable to properly escape. In this situation, equation (40) provides the renovated location of Harris Hawk.

$$M^{T+1} = M^{T}_{prey} - B|\Delta M^{T}| \tag{40}$$

3.6.2.6. Tactic 3: soft encircle with advanced rapid dives. When the targeted prey still has enough energy $|B| \geq 0.5$ to effectively escape and the hunting Harris Hawk continues constructing a soft encircle in the range of $R < 0.5$ thus, this tactic 3 model modifies the location of Harris Hawk. The hunting Harris Hawks must choose the most advantageous dive toward the prey in such a situation which is attained by performing several moves, assessing the new moves according to equation (41), comparing the movement's outcome with that of the previous dive toward the targeted prey, and the levy flight strategy is considered for enhancing the swindling capacity in the way of rapid dives by the group of Harris Hawks if the optimal dive is not determined in the comparison outcomes while attacking the prey and are modeled in equation (42).

$$D = M^{T}_{prey} - B\left|FM^{T}_{prey} - M^{T}\right| \tag{41}$$

$$E = D + V \times LS(W) \tag{42}$$

where W denotes the problem dimension, the random vector is described as V with the size of $1 \times W$. The levy flight strategy is represented as LS and is formulated in equation (43) as follows,

$$LS(M) = \frac{r \times \beta}{|s|^{\frac{1}{\sigma}}}, \beta = \left(\frac{\Gamma(1+\sigma) \times \sin\left(\frac{\pi\sigma}{2}\right)}{\Gamma\left(\frac{1+\sigma}{2}\right) \times \sigma \times 2^{\left(\frac{\sigma-1}{2}\right)}}\right)^{\frac{1}{\sigma}} \tag{43}$$

where the random values inside the range of 0 and 1 are denoted as r and s, the constant is assumed as 1.5 which is represented as β.

Thus, the modified location of the Harris Hawk in the soft encircle with advanced rapid dives is determined by equation (44) as follows,

$$M^{T+1} = \begin{cases} D & if\, f(D) < f(M^{T}) \\ E & if\, f(E) < f(M^{T}) \end{cases} \tag{44}$$

where the optimization problem fitness function is represented as f, utilizing equations (41) and (42), the value of D and E is determined.

3.6.2.7. Tactic 4: hard encircle with advanced rapid dives. At the condition of energy as $|B| < 0.5$ and the random members as $R < 0.5$, there is the occurrence of hard encircle with advanced rapid dives behavior of Harris Hawk. With this tactic, the Harris Hawk is attempting to minimize the typical distance between the position and the targeted rabbit, as opposed to the prior strategy (soft encircle with advanced rapid dives). This tactic is modeled using equation (45) by the hard encircle.

$$M^{T+1} = \begin{cases} D & if\, f(D^{'}) < f(M^{T}) \\ E^{'} & if\, f(E^{'}) < f(M^{T}) \end{cases} \tag{45}$$

where $D^{'}$ is attained using equation (46) as follows, $M_a(T)$ is attained using equation (24), and $E^{'}$ is calculated using equation (47) as follows,

$$D^{'} = M^{T}_{prey} - B\left|FM^{T}_{prey} - M^{T}_{a}\right| \tag{46}$$

$$\mathrm{E}' = D' + V \times LS(W) \tag{47}$$

By utilizing the proposed HHOSSA optimization, the imbalanced data present in the database are well-balanced, and the light GBM as well as the weighted average Bi-LSTM parameters are well-tuned to improve the performance of attack detection. The Bi-LSTM is trained by the proposed HHOSSA optimization by selecting the appropriate parameters involved in the classifier. The parameters, such as learning rate, minimum data in leaf, bagging, and feature fraction are optimally well-tuned by the HHOSSA optimization in the light GBM classifier. The learning rate, minimum batch size, a parameter for dropout regularization, number of neurons in hidden layers, and the number of neural network layers parameters in the Bi-LSTM are well trained by the HHOSSA optimization.

The pseudocode of the HHOSSA optimization is described in algorithm 1 as follows,

S. No — Algorithm 1. HHOSSA optimization-based Bi-LSTM classifier

```
Input: H and $t$;
Output: location of prey and fitness function
Initialization
  $M_j, j = 1, 2, 3, 4, \ldots.., H$
Exploring stage
  Position renovation of Harris Hawk
    If $h \geq 0.5$
    $M^{T+1} = 0.5[M^{T}_{rand}(1 + R_1) + M^{T}(\beta - 2R_1R_2) + M^{T}_{best}(1 - \beta)]$
    Else $h < 0.5$
      $(M_{prey}(T) - M_a(T) - D)$
Shifting from exploring to swindling stage
  If $|B| \geq 1$
    Exploring stage happening
  Else $|B| < 1$
    Swindling stage occurring
Swindling stage
  Tactic 1: soft encircle
    If $|B| \geq 0.5$ and $R \geq 0.5$
  Tactic 2: hard encircle
    If $|B| < 0.5$ and $R \geq 0.5$
  Tactic 3: soft encircle with advanced rapid dives
    If $|B| \geq 0.5$ and $R < 0.5$
  Tactic 4: hard encircle with advanced rapid dives
    If $|B| < 0.5$ and $R < 0.5$
End while
```

4. Results and discussion

The performance of the proposed HHOSSA-hybrid classifier in detecting the attacks is explained in this section, in addition to the relative discussion of various existing methods. The comparative analysis involves the performance of classification, attribute selection, and data balancing of the proposed HHOSSA-hybrid classifier, HHOSSA-based attribute selection, and the HHOSSA-SMOTE algorithm. The performance metrics involved in the performance analysis of the HHOSSA-hybrid classifier are accuracy, sensitivity, and specificity [25].

4.1. Experimental setup

This performance analysis and the comparative performance of the proposed HHOSSA-hybrid classifier are implemented in the Python tool in Windows 10 OS with 8 GB RAM using the DAPT 2020 database.

4.2. Relative methods

The methods used for comparing the performance of the HHOSSA-hybrid classifier are [22,23,26–32], and [33].

4.2.1. Comparative analysis

The performance of the HHOSSA-hybrid classifier in attack detection, the HHOSSA-SMOTE algorithm for data balancing, and HHOSSA-based attribute selection are compared with the various existing methods depending on the value of K-fold and training percentage.

i) Comparison based on K-fold value

The efficiency of the HHOSSA-hybrid classifier with the other existing methods is revealed in Fig. 4. The attained accuracy of the HHOSSA-hybrid classifier and the various existing methods is revealed in Fig. 4 a). The accuracy of the HHOSSA-hybrid classifier in the classification of attacks is 94.468 % with a performance improvement of 4.79 % than the HHO-BiLSTM for the K-Fold value 10.

Fig. 4b) reveals the attained sensitivity of the HHOSSA-hybrid classifier and the various existing methods. The sensitivity of the HHOSSA-hybrid classifier in the classification of attacks is 94.65 % with a performance improvement of 8.7 % than the HHO-BiLSTM for the K-Fold value 10.

Fig. 4c) reveals the attained specificity of the HHOSSA-hybrid classifier and the various existing methods. The specificity of the HHOSSA-hybrid classifier in the classification of attacks is 95.23 % with a performance improvement of 0.89 % than the HHO- BiLSTM for the K-Fold value 10. Thus, the performance of the HHOSSA-hybrid Bi-LSTM attains better performance in classifying the attacks from the provided dataset.

ii) Comparison based on training percentage

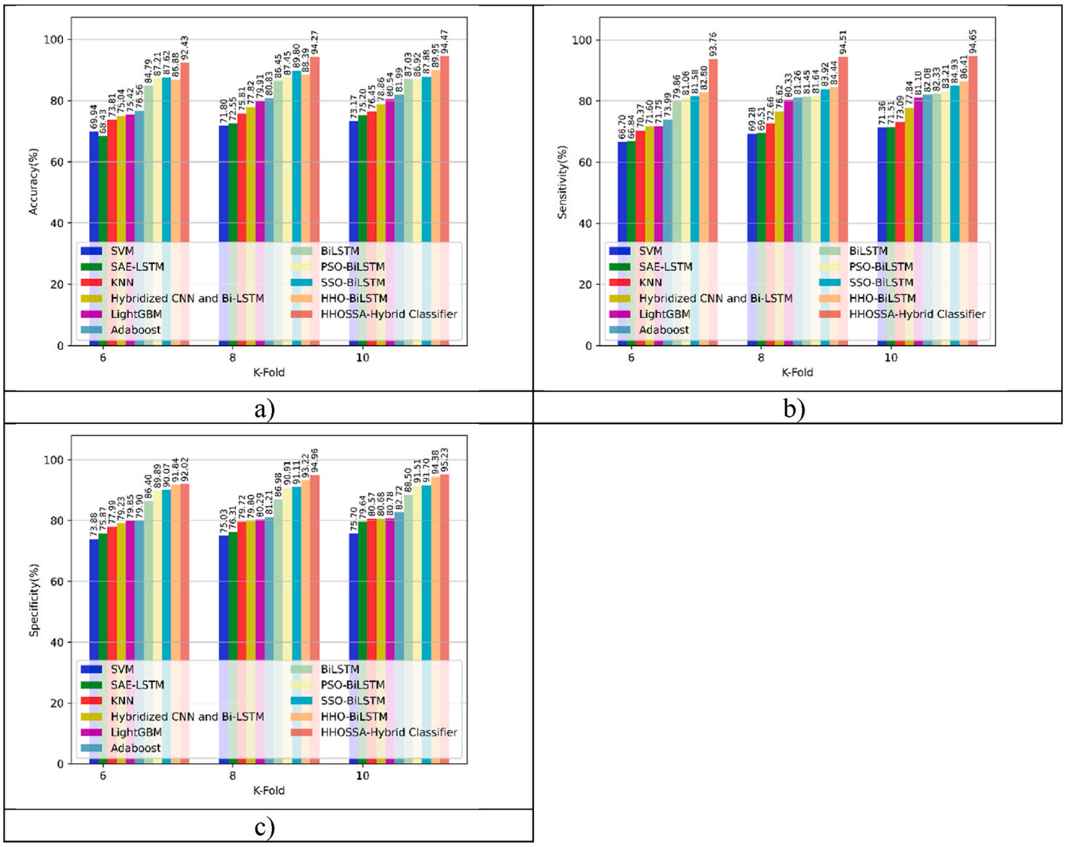
The efficiency of the HHOSSA-hybrid classifier with the other existing methods is revealed in Fig. 5a) which reveals the attained accuracy of the HHOSSA-hybrid classifier and the various existing methods. The accuracy of the HHOSSA-hybrid classifier in the classification of attacks is 96.18 % with a performance improvement of 1.22 % than the HHO-BiLSTM for the training percentage of 80.

Fig. 5b) reveals the attained sensitivity of the HHOSSA-hybrid classifier and the various existing methods. The sensitivity of the HHOSSA-hybrid classifier in the classification of attacks is 96.723 % with a performance improvement of 6.65 % than the HHO-BiLSTM for the 80 % training.

Fig. 5c) reveals the attained specificity of the HHOSSA-hybrid classifier and the various existing methods. The specificity of the HHOSSA-hybrid classifier in the classification of attacks is 96.6 % with a performance improvement of 0.73 % than the HHO- BiLSTM for the 80 % training. Thus, the performance of the HHOSSA-hybrid classifier attains better performance in classifying the attacks from the provided dataset.

4.3. Comparative for data balancing

The existing methods taken into consideration for evaluating the performance of the HHOSSA- SMOTE algorithm in data balancing are No balancing, Random Over sampler [34,35], PSO-SMOTE [36,37], GA-SMOTE [38], SSO-SMOTE [39], and HHO-SMOTE [40].

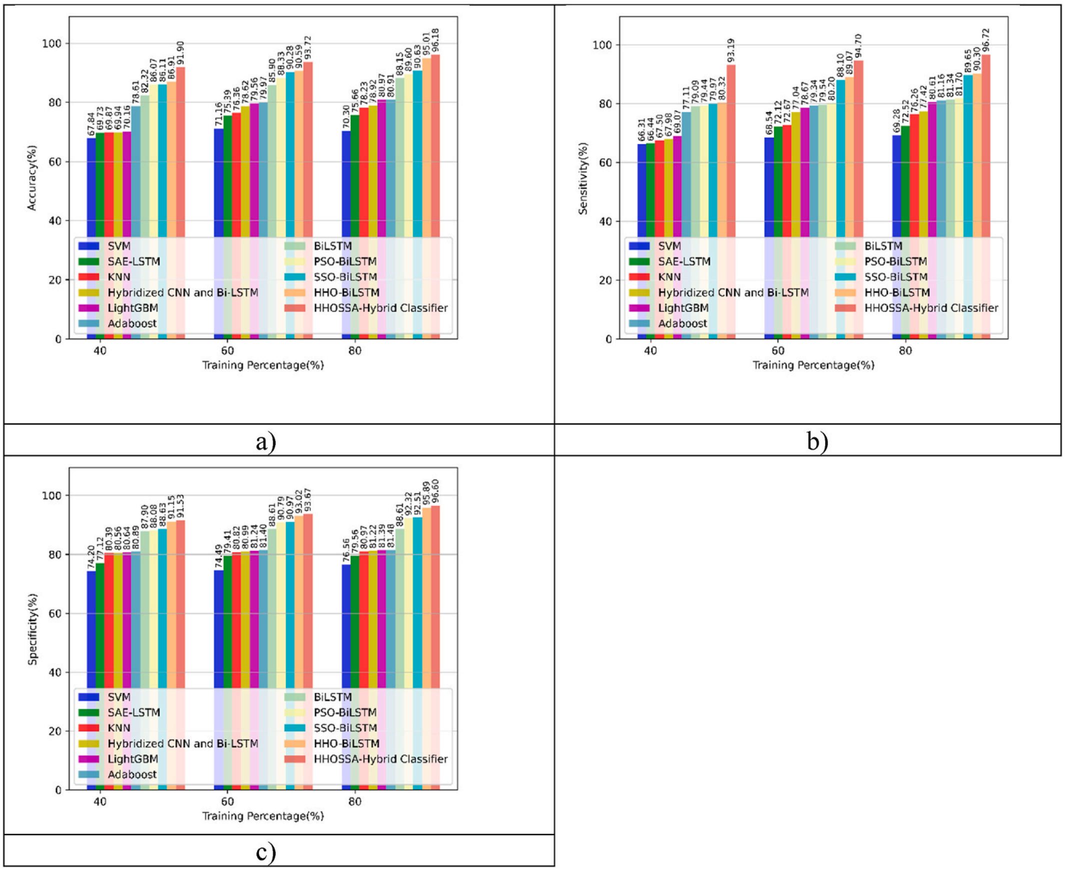


a)

b)

c)

Fig. 4. Comparability based on K-fold value with a) Accuracy b) Sensitivity, and c) Specificity.



a)

b)

c)

Fig. 5. Comparability based on training percentage with a) Accuracy b) Sensitivity, and c) Specificity.

i) Comparative based on training percentage

The efficiency of the HHOSSA-SMOTE algorithm in data balancing is compared with the other existing methods is revealed in Fig. 6a) revealing the attained accuracy of the HHOSSA SMOTE algorithm and the various existing methods. The accuracy of the HHOSSA SMOTE algorithm in the data balancing is 96.038 % with a performance improvement of 0.145 % than the HHO-SMOTE for the 80 % training.

Fig. 6b) reveals the attained sensitivity of the HHOSSA SMOTE algorithm and the various existing methods. The sensitivity of the HHOSSA SMOTE algorithm in the data balancing is 91.400 % with a performance improvement of 0.832 % than the HHO-SMOTE for the 80 % training.

Fig. 6c) reveals the attained specificity of the HHOSSA SMOTE algorithm and the various existing methods. The specificity of the HHOSSA SMOTE algorithm in the data balancing is 97.286 % with a performance improvement of 0.025 % than the HHO-SMOTE for the 80 % training. Thus, the performance of the HHOSSA SMOTE algorithm attains better performance in data balancing for the imbalanced dataset.

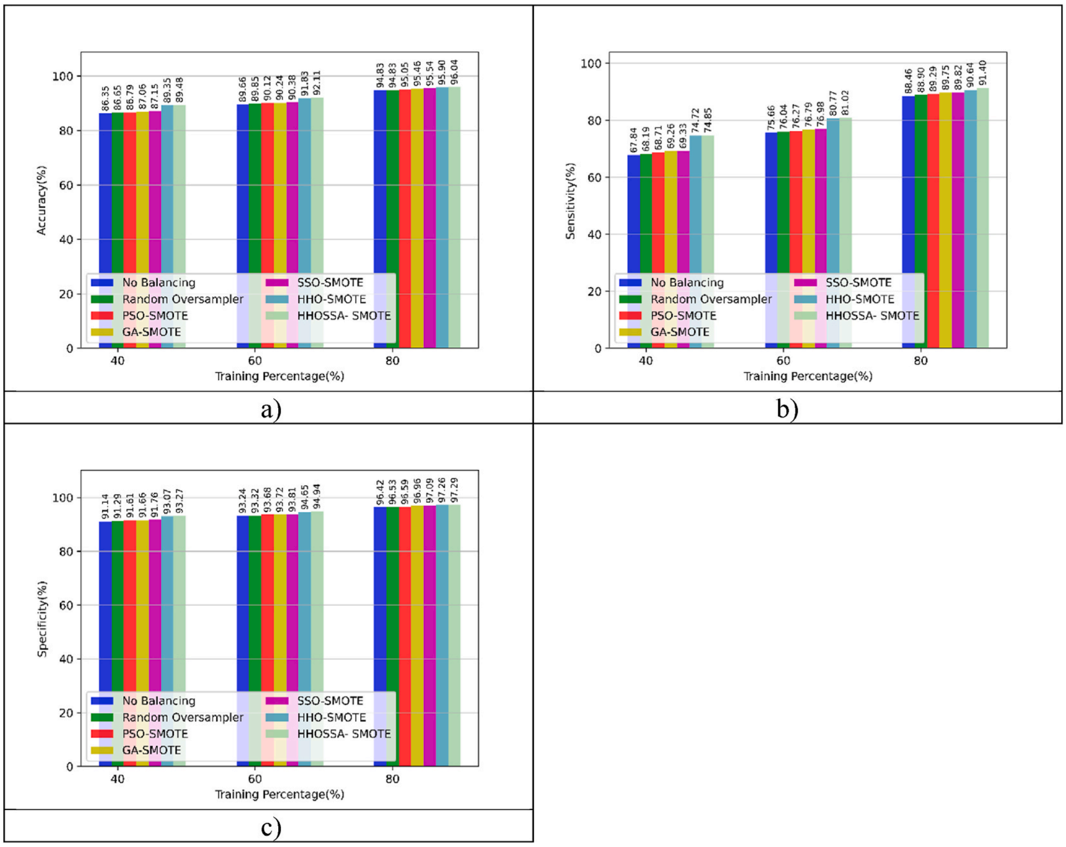
4.4. *Comparative for attribute selection*

The existing methods taken into consideration for evaluating the performance of the HHSSA-based attribute selection are GA-based selection [41], HHO-based selection [42], and SSO-based selection [43].

i) Comparison based on training percentage

The efficiency of the HHOSSA-based attribute selection is compared with the other existing methods is revealed in Fig. 7a) revealing the attained accuracy of the HHOSSA-based attribute selection and the various existing methods. The accuracy of the HHOSSA-based attribute selection is 96.191 % with a performance improvement of 0.102 % than the SSO-based selection for the 80 % training.

Fig. 7b) reveals the attained sensitivity of the HHOSSA-based attribute selection and the various existing methods. The sensitivity of



a)

b)

c)

Fig. 6. Comparative for data balancing a) accuracy b) sensitivity c) specificity.

the HHOSSA-based attribute selection is 90.998 % with a performance improvement of 0.102 % than the SSO-based selection for the 80 % training.

Fig. 7c) reveals the attained specificity of the HHOSSA-based attribute selection and the various existing methods. The specificity of the HHOSSA-based attribute selection is 97.530 % with a performance improvement of 0.102 % than the SSO-based selection for the training percentage of 80. Thus, the performance of the HHOSSA-based attribute selection attains a better performance than the other existing methods.

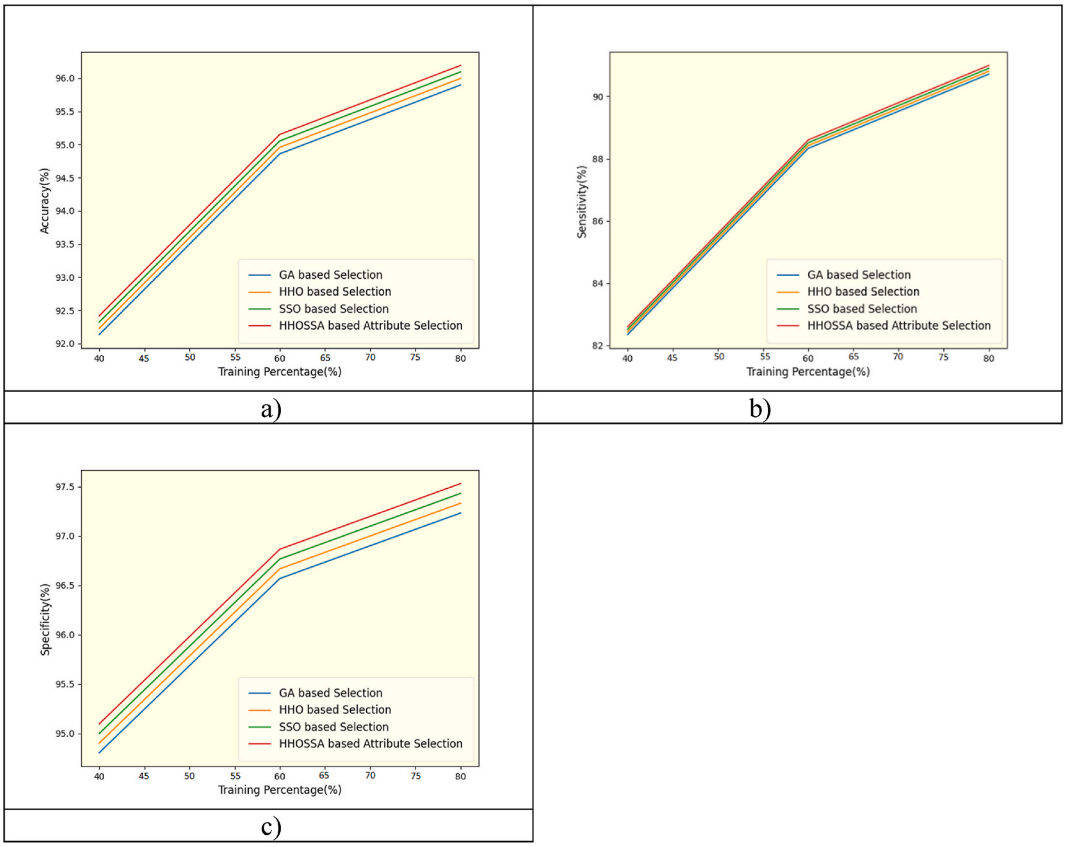
4.5. ROC_AUC analysis

The performance of the HHOSSA-hybrid classifier in classifying the attacks based on the true and false positive rates is revealed in Fig. 8. The sensitivity of the HHOSSA-hybrid classifier is 93.7 % for the minimal 10 % error, while the other existing HHO-BiLSTM classifier attains a sensitivity of 91.5 %. The sensitivity of the HHOSSA-hybrid classifier is improved to 97.9 % for the maximal 90 % error, which is 3.86 % better than the existing HHO-BiLSTM classifier with a sensitivity of 94.1 % [44].

The ROC curve figure showcases the performance of different methods used for APT attack detection. The x-axis represents the various methods, while the y-axis represents the AUC percentage. Starting from the left side of the graph, we can observe the AUC percentages gradually increasing as we move toward the right. Table 1 reveals the SVM method achieves the lowest AUC percentage of 72.435, indicating its relatively lower performance in APT attack detection. As we progress towards the right, we see performance improvements. The SAE-LSTM method achieves an AUC percentage of 74.441, followed by the KNN method with 77.157. The Hybridized CNN and Bi-LSTM method show further improvement, reaching an AUC percentage of 79.218.

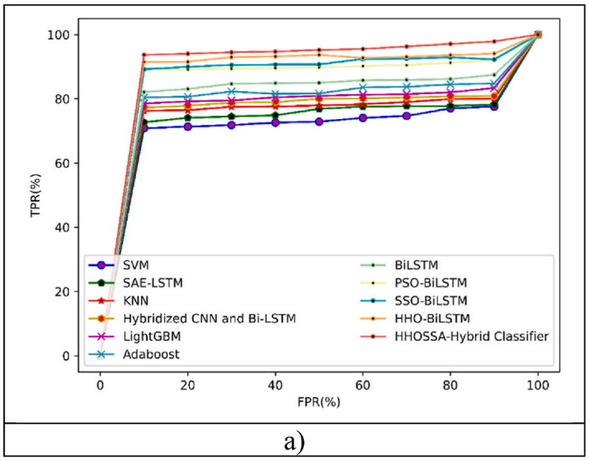
Continuing, we see the Light GBM method performing even better with an AUC percentage of 80.518, followed by the Adaboost method with 82.141. The BiLSTM method demonstrates a higher AUC percentage of 83.861, indicating its stronger performance in APT attack detection. Moving toward the right side of the graph, we observe significant performance improvements. The PSO-BiLSTM method achieves an AUC percentage of 85.452, followed by the SSO-BiLSTM method with 87.373. The HHO-BiLSTM method demonstrates a notable improvement, reaching an AUC percentage of 91.751.

Finally, the HHOSSA - Hybrid classifier showcases the highest AUC percentage of 97.032, highlighting its exceptional performance in detecting APT attacks. This method surpasses all others in terms of AUC, indicating its effectiveness and superiority for APT attack detection [44]. Overall, the AUC curve figure clearly illustrates the comparative performance of different methods, with the HHOSSA -



a)
b)
c)

Fig. 7. Comparative for attribute selection a) accuracy b) sensitivity c) specificity.



a)

Fig. 8. ROC_AUC of the HHOSSA-hybrid classifier with other existing methods.

Hybrid classifier standing out as the top-performing method in APT attack detection.

4.6. PRC_AUC analysis

The PRC_AUC analysis of the HHOSSA hybrid BiLSTM and the existing methods are revealed in Fig. 9. The five different attacks include benign traffic, reconnaissance, established foothold, lateral movement, and data exfiltration. The efficient classification performance of the method most significantly depends on the precision and recall value.

Table 1
AUC of the HHOSSA-hybrid classifier and other existing methods based on ROC.

Methods	AUC %
SVM	72.435
SAE-LSTM	74.441
KNN	77.157
Hybridized CNN and Bi-LSTM	79.218
Light GBM	80.518
Adaboost	82.141
BiLSTM	83.861
PSO-BiLSTM	85.452
SSO-BiLSTM	87.373
HHO-BiLSTM	91.751
HHOSSA -Hybrid classifier	**97.032**

Fig. 9a) reveals the performance of the HHOSSA-hybrid classifier during benign traffic, which attains a precision of 80.737 % with an accuracy improvement of 2.76 % than the existing HHO-BiLSTM classifier for the sensitivity of 50 %. It outperforms all other methods listed, including SVM (AUC% 64.081), SAE-LSTM (AUC% 65.293), KNN (AUC% 67.003), and others. This significant difference in AUC% values indicates that the HHOSSA-Hybrid classifier exhibits a higher level of accuracy and precision in distinguishing benign traffic from potentially malicious activity. Its ability to correctly identify normal network behavior ensures fewer false positives and provides a solid foundation for reliable threat detection [44].

Fig. 9b) reveals the performance of the HHOSSA-hybrid classifier during the reconnaissance, which attains a precision of 81.096 % with an accuracy improvement of 4.51 % than the existing HHO-BiLSTM classifier for the sensitivity of 50 %. It outperforms SVM (62.276), SAE-LSTM (62.939), KNN (64.939), and other algorithms. This indicates that the HHOSSA-Hybrid classifier effectively identifies and thwarts initial probing and information gathering by potential attackers. Its higher AUC% value demonstrates its superior ability to detect reconnaissance activities, allowing for early detection and proactive defense measures to be implemented.

Fig. 9c) reveals the performance of the HHOSSA-hybrid classifier during the established foothold, which attains a precision of 79.097 % with an accuracy improvement of 0.91 % than the existing HHO-BiLSTM classifier for the sensitivity of 50 %. This outperforms SVM (64.960), SAE-LSTM (66.715), KNN (68.097), and other methods. The higher AUC% value indicates that the HHOSSA-Hybrid classifier is more accurate in detecting and mitigating attempts by attackers to gain persistent access within a network. By effectively identifying the presence of established footholds, the classifier aids in prompt response and containment of potential threats, bolstering the overall security posture of the system.

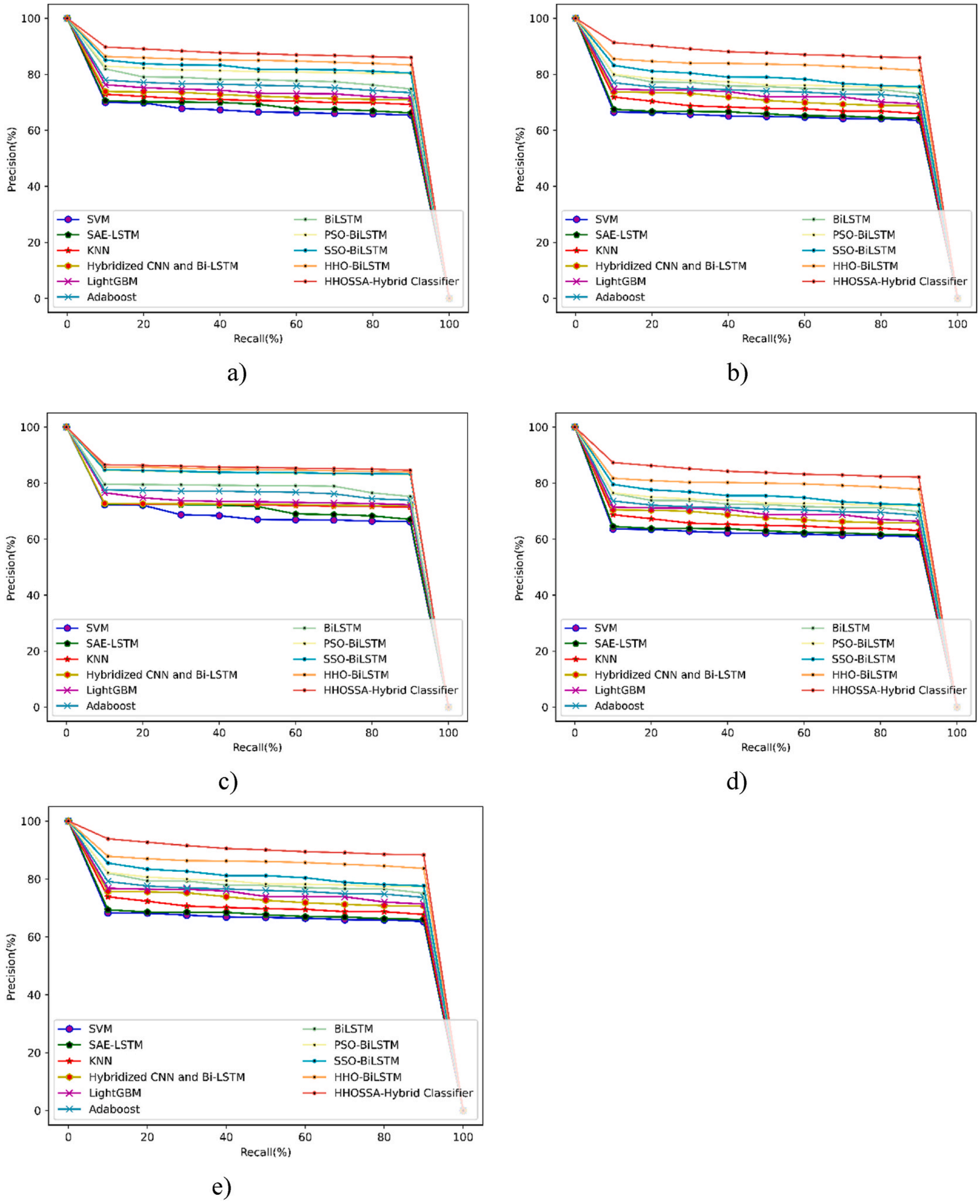
Fig. 9d) reveals the performance of the HHOSSA-hybrid classifier during the lateral movements, which attains a precision of 77.914 % with an accuracy improvement of 4.51 % than the existing HHO-BiLSTM classifier for a sensitivity of 50 %. It Outperforms SVM (59.925), SAE-LSTM (60.560), KNN (62.471), and other methods. The significant difference in AUC% values demonstrate the classifier's effectiveness in detecting and preventing unauthorized lateral movement attempts. Its ability to accurately identify suspicious activities related to lateral movement enables swift response and containment, minimizing the potential impact of intrusions and reducing the risk of lateral spread of attacks.

Fig. 9e) reveals the performance of the HHOSSA-hybrid classifier during the data exfiltration, which attains a precision of 83.094 % with an accuracy improvement of 4.51 % than the existing HHO-BiLSTM classifier for the sensitivity of 50 %. Surpassing the AUC% values of SVM (63.751), SAE-LSTM (64.434), KNN (66.489), and other methods. This indicates that the HHOSSA-Hybrid classifier is highly effective in detecting and preventing unauthorized attempts to extract sensitive data. Its advanced classification capabilities enable it to identify patterns and anomalies associated with data exfiltration, reducing the risk of data breaches and enhancing overall network security.

In summary, Table 2 the HHOSSA-hybrid classifier consistently achieves higher AUC% values across all attack categories, surpassing the performance of other algorithms such as SVM, SAE-LSTM, KNN, and others. Its advanced classification performance, as indicated by the higher AUC% values in the PRC_AUC analysis, makes it the preferred choice for detecting and mitigating network attacks. The HHOSSA-hybrid classifier's enhanced accuracy, precision, and ability to detect various attack types make it a powerful tool for ensuring robust network security and safeguarding against evolving threats. From all the compared methods, the HHOSSA-hybrid classifier attains the greater AUC based on the PRC_AUC analysis.

4.7. Comparative discussion

The HHOSSA-Hybrid Classifier is evaluated by considering the performance measures as accuracy, sensitivity, and specificity based on the value of K-fold and training percentage using the DAPT 2020 dataset. Table 3 reveals the overall performance of the HHOSSA-Hybrid Classifier with their performance metrics. Variable length sequences can be handled using Bi-LSTMs. They don't need fixed-size inputs since they can efficiently handle sequences of any length. This adaptability is especially useful for tasks where the total length of the sequence fluctuates, such as natural language processing applications that require processing phrases of various lengths. Unlike several other gradient boosting methods, Light GBM is intended to be extremely effective and quick. Instead of the level-wise technique employed by conventional gradient boosting algorithms, it employs a leaf-wise tree growth strategy. When working with huge



a)

b)

c)

d)

e)

Fig. 9. PRC_AUC for a) Benign traffic b) Reconnaissance c) Established foothold d) Lateral movements e) Data exfiltration.

datasets, this strategy can result in quicker training times and improved scalability.

Table 4 reveals the HHOSSA-Hybrid Classifier attains a classification accuracy of 80.737 % of Benign traffic, 83.094 % of Data exfiltration, 79.097 % of Established foothold, 77.914 % of Lateral movements and 81.096 % of Reconnaissance with all APT attack phases. The HHOSSA-Hybrid Classifier attains a classification accuracy of 94.468 %, a sensitivity of 94.65 %, and a specificity of 95.23 %with the K-Fold value of 10. The HHOSSA-Hybrid Classifier attains a classification accuracy of 96.18 %, a sensitivity of 96.723 %, and a specificity of 96.6 % with the training percentage.

5. Conclusion

In this research, the initial step of the quasi-identifier discovery is carried out by selecting the most significant attributes from the provided dataset using the proposed HHOSSA-based attribute selection method. The irrelevant data attributes present in the dataset

Table 2
AUC of the proposed HHOSSA Hybrid Classifier and other existing methods based on PRC.

Methods	Attacks				
	Benign traffic	Data exfiltration	Established foothold	Lateral movements	Reconnaissance
	AUC %				
SVM	64.081	63.751	64.960	59.925	62.276
SAE-LSTM	65.293	64.434	66.715	60.560	62.939
KNN	67.003	66.489	68.097	62.471	64.939
Hybridized CNN and Bi-LSTM	68.270	68.863	68.269	64.679	67.249
Light GBM	69.384	70.044	69.328	65.777	68.398
Adaboost	71.159	71.374	71.564	67.014	69.692
Bi-LSTM	72.905	72.873	73.291	68.409	71.151
PSO-BiLSTM	75.536	73.746	77.588	69.220	72.000
SSO-BiLSTM	76.558	75.342	77.714	70.704	73.553
HHO-BiLSTM	78.554	79.301	78.501	74.386	77.405
HHOSSA-Hybrid Classifier	**80.737**	**83.094**	**79.097**	**77.914**	**81.096**

Table 3
Performance of the HHOSSA Hybrid Classifier over other existing methods.

Methods	DAPT 2020 database					
	K-Fold 10			Training percentage 80		
	Accuracy %	Sensitivity %	Specificity %	Accuracy %	Sensitivity %	Specificity %
SVM	73.167	71.36	75.705	70.299	69.279	76.555
SAE-LSTM	75.197	71.506	79.64	75.661	72.517	79.561
KNN	76.445	73.088	80.567	78.225	76.261	80.972
Hybridized CNN and Bi-LSTM	78.865	77.839	80.679	78.923	77.419	81.216
Light GBM	80.537	81.104	80.776	80.97	80.614	81.389
Adaboost	81.988	82.077	82.719	80.913	81.156	81.479
BiLSTM	87.034	82.327	88.496	88.147	81.337	88.614
PSO-BiLSTM	86.924	83.206	91.512	89.605	81.699	92.325
SSO-BiLSTM	87.875	84.93	91.699	90.627	89.646	92.514
HHO-BiLSTM	89.946	86.413	94.379	95.011	90.295	95.891
HHOSSA -Hybrid classifier	**94.468**	**94.65**	**95.23**	**96.18**	**96.723**	**96.60**

Table 4
The performance of the HHOSSA -Hybrid Classifier.

HHOSSA -Hybrid Classifier		
Attacks	Benign traffic	**80.737**
	Data exfiltration	**83.094**
	Established foothold	**79.097**
	Lateral movements	**77.914**
	Reconnaissance	**81.096**
K-Fold (%)	Accuracy	**94.460**
	Sensitivity	**94.65**
	Specificity	**95.23**
TP (%)	Accuracy	**96.18**
	Sensitivity	**96.72**
	Specificity	**96.60**

are eliminated and then integrated with the time domain-based statistical features, as well as the data, attributes to enhance the performance of the better classification. The unbalanced data, such as the lateral movements and the data exfiltration in the DAPT 2020 database are successfully balanced by the HHOSSA-SMOTE method, which further increases the performance of the classifier. The developed HHOSSA optimization is well-tuned with the light GBM and Bi-LSTM classifier hyperparameters for the accurate classification of the attacks. The final detection of the attacks existing in the DAPT database is generated by the output of both the optimized light GBM and the Bi-LSTM classifier. The proposed HHOSSA-hybrid classifier attains performance improvement of 4.79 %, 8.70 %, and 0.89 % when compared to the HHO-BiLSTM classifier with an accuracy of 94.468 %, the sensitivity of 94.650 %, specificity of 95.230 % for the K-fold value 10. The limitation of this research is the lack of real-world evaluation. Therefore, future work should focus on conducting rigorous evaluations using real-world data to assess the performance and practical applicability of the proposed HHOSSA approach. As every organization has started adopting cloud services there is a need to adapt for APT detection to find cloud-based detection techniques hence the cloud-based APT detection scheme will be developed in the future. Hence, In the future, a model that automatically adapts to various environments and traffic will be developed with large datasets.

Funding

This research was supported by Korea Institute of Science and Technology Information (KISTI). (No. (KISTI) K-23-L04-C05, (NTIS) 1711198593)

CRediT authorship contribution statement

Indra Kumari: Formal analysis, Investigation, Resources, Validation, Writing – original draft. **Minho Lee:** Conceptualization, Data curation, Project administration, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was supported by Korea Institute of Science and Technology Information (KISTI), University of Science and Technology (UST), Republic of Korea and Prof. Jungsuk Song owe thanks for the support and excellent research facilities.

References

[1] C. Do Xuan, M.H. Dao, A novel approach for APT attack detection based on combined deep learning model, Neural Comput. Appl. 33 (20) (2021) 13251–13264, https://doi.org/10.1007/s00521-021-05952-5.

[2] S. Myneni, et al., DAPT 2020 - Constructing a Benchmark Dataset for Advanced Persistent Threats, 2020, pp. 138–163, https://doi.org/10.1007/978-3-030-59621-7_8.

[3] S. Myneni, et al., Unraveled — a semi-synthetic dataset for advanced persistent threats, Comput. Networks 227 (May 2023), 109688, https://doi.org/10.1016/j.comnet.2023.109688.

[4] A.B. Ajmal, M.A. Shah, C. Maple, M.N. Asghar, S.U. Islam, Offensive security: towards proactive threat hunting via adversary emulation, IEEE Access 9 (2021) 126023–126033, https://doi.org/10.1109/ACCESS.2021.3104260.

[5] H. Haddadpajouh, A. Azmoodeh, A. Dehghantanha, R.M. Parizi, MVFCC: a multi-view fuzzy consensus clustering model for malware threat attribution, IEEE Access 8 (2020) 139188–139198, https://doi.org/10.1109/ACCESS.2020.3012907.

[6] T. Bai, H. Bian, M.A. Salahuddin, A. Abou Daya, N. Limam, R. Boutaba, RDP-Based lateral movement detection using machine learning, Comput. Commun. 165 (2021) 9–19, 10.1016/j.comcom.2020.1 0.013.

[7] "DAPT 2020 dataset" [Online]. Available: https://gitlab.thothlab.org/achaud16/apt/-/tree/master/csv.

[8] H.M. Alabool, D. Alarabiat, L. Abualigah, A.A. Heidari, Harris Hawks optimization: a comprehensive review of recent variants and applications, Springer London 33 (15) (2021), https://doi.org/10.1007/s00521-021-05720-5.

[9] J. Xue, B. Shen, A novel swarm intelligence optimization approach: sparrow search algorithm, Syst. Sci. Control Eng. 8 (1) (2020) 22–34, https://doi.org/10.1080/21642583.2019.1708830.

[10] A. Alshamrani, S. Myneni, A. Chowdhary, D. Huang, A survey on advanced persistent threats: techniques, solutions, challenges, and research opportunities, IEEE Commun. Surv. Tutorials 21 (2) (2019) 1851–1877, https://doi.org/10.1109/COMST.2019.2891891.

[11] S. Sengupta, A. Chowdhary, D. Huang, S. Kambhampati, General Sum Markov Games for Strategic Detection of Advanced Persistent Threats Using Moving Target Defense in Cloud Networks, 2019, pp. 492–512, https://doi.org/10.1007/978-3-030-32430-8_29.

[12] F. Wang, R. Li, Z. Zhang, APTSID: an ensemble learning method for APT attack stage identification, in: 2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT), IEEE, Oct. 2021, pp. 190–195, https://doi.org/10.1109/ACAIT53529.2021.9731169.

[13] M. Alrehaili, A. Alshamrani, A. Eshmawi, A hybrid deep learning approach for advanced persistent threat attack detection, in: The 5th International Conference on Future Networks & Distributed Systems, ACM, New York, NY, USA, Dec. 2021, pp. 78–86, https://doi.org/10.1145/3508072.3508085.

[14] T. Savaş, S. Savaş, Tekdüzen kaynak bulucu yoluyla kimlik avı tespiti için makine öğrenmesi algoritmalarının özellik tabanlı performans karşılaştırması, J. Polytech. (3) (2022) 1261–1270, https://doi.org/10.2339/politeknik.1035286, 0900.

[15] S. Savaş, S. Karataş, Cyber governance studies in ensuring cybersecurity: an overview of cybersecurity governance, Int. Cybersecurity Law Rev. 3 (1) (2022) 7–34, https://doi.org/10.1365/s43439-021-00045-4. Jun.

[16] W. Meng, X. Luo, W. Li, Y. Li, Design and evaluation of advanced collusion attacks on collaborative intrusion detection networks in practice, in: 2016 IEEE Trustcom/BigDataSE/ISPA, IEEE, Aug. 2016, pp. 1061–1068, https://doi.org/10.1109/TrustCom.2016.0176.

[17] A. Dijk, Detection of advanced persistent threats using artificial intelligence for deep packet inspection, in: 2021 IEEE International Conference on Big Data (Big Data), IEEE, Dec. 2021, pp. 2092–2097, https://doi.org/10.1109/BigData52589.2021.9671464.

[18] J.R. Moya, N. DeCastro-García, R.-Á. Fernández-Díaz, J.L. Tamargo, Expert knowledge and data analysis for detecting advanced persistent threats, Open Math. 15 (1) (Aug. 2017) 1108–1122, https://doi.org/10.1515/math-2017-0094.

[19] I. Singh, N. Kumar, S. K.G, T. Sharma, V. Kumar, S. Singhal, Database intrusion detection using role and user behavior based risk assessment, J. Inf. Secure. Appl. 55 (October) (2020), 102654, https://doi.org/10.1016/j.jisa.2020.102654.

[20] M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, N.K. Jha, Systematic poisoning attacks on and defenses for machine learning in healthcare, IEEE J. Biomed. Heal. Informatics 19 (6) (2015) 1893–1905, https://doi.org/10.1109/JBHI.2014.2344095.

[21] J. Singh, N. Jyoti, S. Behal, On the use of information theory metrics for detecting DDoS attacks and flash events: an empirical analysis, comparison, and future directions, Kuwait J. Sci. 48 (4) (2021) 1–24, https://doi.org/10.48129/KJS.V48I4.10612.

[22] J. Fan, X. Ma, L. Wu, F. Zhang, X. Yu, W. Zeng, Light Gradient Boosting Machine: an efficient soft computing model for estimating daily reference evapotranspiration with local and external meteorological data, Agric. Water Manag. 225 (August) (2019), 105758, https://doi.org/10.1016/j.agwat.2019.105758.

[23] J. Yang, M. Zhou, B. Cui, MLAB-BiLSTM: Online Web Attack Detection via Attention-Based Deep Neural Networks, 2020, pp. 482–492, https://doi.org/10.1007/978-981-15-9129-7_33.

[24] D. Binu, B.S. Kariyappa, Rider-deep-lstm network for hybrid distance score-based fault prediction in analog circuits, IEEE Trans. Ind. Electron. 68 (10) (2021) 10097–10106, https://doi.org/10.1109/TIE.2020.3028796.

[25] J.H. Joloudari, H. Saadatfar, A. Dehzangi, S. Shamshirband, Computer-aided decision-making for predicting liver disease using PSO-based optimized SVM with feature selection, Informat. Med. Unlocked 17 (2019).

[26] J. Ye, X. Cheng, J. Zhu, L. Feng, L. Song, A DDoS attack detection method based on SVM in software defined network, Secur. Commun. Networks 2018 (2018), https://doi.org/10.1155/2018/9804061.

[27] K.U. Jaseena, B.C. Kovoor, A hybrid wind speed forecasting model using stacked autoencoder and LSTM, J. Renew. Sustain. Energy 12 (2) (Mar. 2020), 023302, https://doi.org/10.1063/1.5139689.

[28] S. Dong, M. Sarem, DDoS attack detection method based on improved KNN with the degree of DDoS attack in software-defined networks, IEEE Access 8 (2020) 5039–5048, https://doi.org/10.1109/ACCESS.2019.2963077.

[29] G. Deepak, S. Rooban, A. Santhanavijayan, A knowledge-centric hybridized approach for crime classification incorporating deep bi-LSTM neural network, Multimed. Tools Appl. 80 (18) (2021) 28061–28085, https://doi.org/10.1007/s11042-021-11050-4.

[30] D. Tang, L. Tang, R. Dai, J. Chen, X. Li, J.J.P.C. Rodrigues, MF-Adaboost: LDoS attack detection based on multi-features and improved Adaboost, Futur. Gener. Comput. Syst. 106 (2020) 347–359, https://doi.org/10.1016/j.future.2019.12.034.

[31] G. Zhang, F. Tan, Y. Wu, Ship motion attitude prediction based on an adaptive dynamic particle swarm optimization algorithm and bidirectional LSTM neural network, IEEE Access 8 (2020) 90087–90098, https://doi.org/10.1109/ACCESS.2020.2993909.

[32] A.K. Nandanwar, J. Choudhary, Semantic features with contextual knowledge-based web page categorization using the GloVe model and stacked BiLSTM, Symmetry (Basel) 13 (10) (Sep. 2021) 1772, https://doi.org/10.3390/sym13101772.

[33] H.M. Balaha, E.M. El-Gendy, M.M. Saafan, CovH2SD: a COVID-19 detection approach based on Harris Hawks Optimization and stacked deep learning, Expert Syst. Appl. 186 (June) (2021), 115805, https://doi.org/10.1016/j.eswa.2021.115805.

[34] H.N. Eke, A. Petrovski, Advanced persistent threats detection based on deep learning approach, in: 2023 IEEE 6th International Conference on Industrial Cyber-Physical Systems (ICPS), IEEE, May 2023, pp. 1–10, https://doi.org/10.1109/ICPS58381.2023.10128062.

[35] A. Moreo, A. Esuli, F. Sebastiani, Distributional random oversampling for imbalanced text classification, SIGIR 2016 - Proc. 39th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr. (2016) 805–808, https://doi.org/10.1145/2911451.2914722.

[36] S. Susan, A. Kumar, Hybrid of Intelligent Minority Oversampling and PSO-Based Intelligent Majority Undersampling for Learning from Imbalanced Datasets, 2020, pp. 760–769, https://doi.org/10.1007/978-3-030-16660-1_74.

[37] M. Gao, X. Hong, S. Chen, C.J. Harris, A combined SMOTE and PSO based RBF classifier for two-class imbalanced problems, Neurocomputing 74 (17) (Oct. 2011) 3456–3466, https://doi.org/10.1016/j.neucom.2011.06.010.

[38] K. Jiang, J. Lu, K. Xia, A novel algorithm for imbalance data classification based on genetic algorithm improved SMOTE, Arab. J. Sci. Eng. 41 (8) (2016) 3255–3266, https://doi.org/10.1007/s13369-016-2179-2.

[39] S. Susan, A. Kumar, The balancing trick: an optimized sampling of imbalanced datasets—a brief survey of the recent State of the Art, Eng. Reports 3 (4) (2021), https://doi.org/10.1002/eng2.12298.

[40] E.H. Houssein, Z. Abohashima, M. Elhoseny, W.M. Mohamed, An Efficient Binary Harris Hawks Optimization Based on Quantum SVM for Cancer Classification Tasks, 2022, pp. 247–258, https://doi.org/10.1049/icp.2021.2680.

[41] J. Liu, et al., Adaptive intrusion detection via GA-GOGMM-based pattern learning with fuzzy rough set-based attribute selection, Expert Syst. Appl. 139 (2020), https://doi.org/10.1016/j.eswa.2019.112845.

[42] K. Dev, P.K.R. Maddikunta, T.R. Gadekallu, S. Bhattacharya, P. Hegde, S. Singh, Energy optimization for green communication in IoT using Harris Hawks optimization, IEEE Trans. Green Commun. Netw. 6 (2) (Jun. 2022) 685–694, https://doi.org/10.1109/TGCN.2022.3143991.

[43] C. Bae, W.C. Yeh, N. Wahid, Y.Y. Chung, Y. Liu, A new simplified swarm optimization (SSO) using exchange local search scheme, Int. J. Innov. Comput. Inf. Control 8 (6) (2012) 4391–4406.

[44] Y. Xiao, L. Liu, Z. Ma, Z. Wang, W. Meng, Defending co-resident attack using reputation-based virtual machine deployment policy in cloud computing, Trans. Emerg. Telecommun. Technol. 32 (9) (Sep. 2021), https://doi.org/10.1002/ett.4271.