Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

A Consistent Augmented Stacking Polynomial Optimized Tool (ASPOT) for Improving Security of Cloud-IoT Systems

Divya Ramachandran[1,*], R. Chithambaramani[2], S. Sankar Ganesh[3], M. Dilli Babu [4]

[1] Department of Information Technology, PSNA College of Engineering and Technology, Dindigul, 624622 Tamil Nadu, India
[2] Department of CSE, School of Computing, Vel Tech Rangarajan Dr Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, 600062 Tamil Nadu, India
[3] Department of Computer Science and Engineering, Kommuri Pratap Reddy Institute of Technology, Ghanpur, Medchal, 501301 Malkajgiri Telangana, India
[4] Department of information Technology, Panimalar engineering college, Poonamallee, Chennai-600123, Tamil Nadu, India

ARTICLE INFO

Article history:
Received 3 August 2023
Received in revised form 17 November 2023
Accepted 30 November 2023
Available online 9 January 2024

Keywords:

Cloud; Internet of Things (IoT); Security; Intrusion detection system (IDS); Deep learning; Augmented stacking polynomial optimized tool (ASPOT); Binary sand cat swarm optimization (BSCSO)

ABSTRACT

The cloud computing transforms information technology by offering end users simulated, flexible resources on demand that require fewer resources and facilities giving them greater flexibility. These materials are delivered over the Internet using predefined networking protocols, regulations, and styles, and they are overseen by various management groups. There are flaws and vulnerabilities in the underlying technology and legacy protocols that could allow an attacker to get access. A recent assessment of the literature led to the conclusion that most intelligence algorithms have a number of complex issues, including a high false prediction rate, difficulty classifying threats, high processing costs, and system load. Hence, the proposed work aims to develop an innovative and lightweight Augmented Stacking Polynomial Optimized Tool (ASPOT) for strengthening the cloud-IoT system security against modern cyberattacks with a accuracy of 99%. The current study uses the lightweight Deep Augmented Preprocessing Model (DAPM) to clean and normalize the input cyber-attack dataset by executing transformation and normalization operations on it. Furthermore, the Binary Sand Cat Swarm Optimization (BSCSO) technique is utilized to identify the most significant and relevant features of the normalized dataset in an optimal manner. Moreover, the class of assaults is promptly and precisely identified from the given data by applying the Deep Stacking Polynomial Learning Network (DSPLN) technology. The effectiveness and results of the proposed ASPOT method are analysed with the use of current cyber-attack statistics and a variety of assessment metrics.

1. Introduction

A particular kind of internet-based computing known as "cloud computing" offers a centralized repository of resources, including memory, internet bandwidth, computing power, and applications created by users [1,2]. With reduced maintenance and building costs, these materials can be easily

* *Corresponding author.*
E-mail address: drdivyarphd@gmail.com

https://doi.org/10.37934/araset.37.1.1636

and immediately delivered to final users over the Internet. Over time, cloud computing has grown significantly. Virtual machines are more affordable and offer greater versatility and upon request, ubiquitous computing capabilities than authentic machines [3-5]. Users utilize cloud-based virtual computers that they acquire to do tasks. Certain tasks are difficult to complete on personal devices due to their high computational complexity. In an effort to reduce power consumption and computation time, portable devices like notebooks and smart phones also store their intensive computing tasks in the cloud. A cloud infrastructure based on the Internet of Things (IoT) [6] is a vast network that has multiple IoT-enabled devices and services. Servers, also known storage spaces, underpinning facilities instantaneous processing, and operations are all included in the framework [7,8]. In addition, standards and services required for interacting, managing, and protecting various IoT devices as well as applications constitute a component of an IoT-based cloud system. For many industrial applications, cloud computing offers regular hardware as well as software updates together with adaptability. Furthermore, the cloud offers a variety of security solutions and lets the user utilize network resources efficiently. Despite these benefits, it is clear that cloud computing has a lot of potential [9,10]. Future developments in cloud computing and its supporting technologies could lead to a plethora of new applications, strategies, products and services, systems, and other potential for enterprises.

Machines are becoming increasingly necessary to evaluate the massive amounts of data people produce and constitute more precise conclusions as the amount of data we produce rises exponentially [11,12]. In light of this, increasing the capacity of devices to deal with, assess, and derive information from huge volumes of data also needs more attention. It is necessary for methods of decision-making to change as data volumes increase. Ensuring the optimal experience for cloud supervisors, software developers, and end users is of the utmost importance for the triumph of any internet-based system [34]. Adoption of clouds is hampered by a number of unique issues, including complexity, control, security, confidentiality, reliability, and costs [13,14]. Depending on the cloud service model selected, data and applications may reside at several tiers, making security in cloud-IoT an essential barrier. Because of this ambiguity, scholars believe that security is the main issue with IoT-cloud computing [35]. The development of the internet and the usage of computers has led to a massive electronic transformation of data, which has raised a number of issues with privacy, confidentiality, and security of information. Some of the cloud threats are Information breach, Information loss, denial of service, Malicious injection, Man in the Middle attack, Phishing, Shared Technology, Account hijacking [31]. The security of computing devices has improved significantly in recent years [15]. On the other hand, potential serious issues with computing devices include safety, confidentiality, and secrecy regarding computer systems. In actuality, there isn't an architecture in existence today that is completely secure. A group of computing devices linked together with the intention of sharing resources is referred to as a network of computers [32,33].

Based on a recent literature review [16-18], it can be deduced that most intelligence algorithms face various complexities and challenges associated with high false prediction rate, low recognition while classifying the attacks, high computation cost, and system burden. Therefore, the goal of the proposed research project is to put into practice a cutting-edge security algorithm to improve cloud-IoT system security. The following lists the main research goals that motivated this work:

i. The Augmented Stacking Polynomial Optimized Tool (ASPOT), an intuitive and novel security methodology that improves cloud-IoT system security against contemporary cyberattacks.
ii. The present study employs the lightweight Deep Augmented Preprocessing Model (DAPM) to execute transformation and normalization operations on the input cyber-

attack dataset, thereby cleaning and normalizing it.

iii. Additionally, the normalized dataset's most significant and relevant features are optimally selected using the Binary Sand Cat Swarm Optimization (BSCSO) algorithm.

iv. Moreover, the Deep Stacking Polynomial Learning Network (DSPLN) methodology is used in order to quickly and accurately determine the class of incursions from the provided data.

v. The performance and outcomes of the suggested ASPOT approach are examined using a range of evaluation metrics in conjunction with contemporary cyber-attack statistics.

The remaining sections of this work are divided into the following categories: In Section 2, a review of various current studies in the field of cloud-IoT security is conducted. The benefits and drawbacks of each approach are examined based on the effectiveness of intrusion detection. In Section 3, the flow model and algorithms are presented together with a thorough discussion of the suggested security solution. In Section 4, the effectiveness and outcomes of the suggested security methodology are evaluated and validated using a variety of performance metrics and publicly available statistics. Lastly, in Section 5, the paper's general summary is given together with the conclusions, results, and outcomes.

2. Related Works

This section looks at a number of current security techniques for protecting cloud-IoT systems, and it discusses the difficulties and issues that come with using the traditional methods.

Abd-Elaziz, *et al.,* [19] deployed the Capuchin Searching Algorithm (CapSa) to defend Internet of Things systems against malicious attacks. For the purpose of this study, the deep neural network classification technique is utilized to maximize the accuracy of classifying normal and invasive occurrences. Additionally, four different datasets are used in this study to evaluate the effectiveness of the recommended strategy. Using the DNN-CapSa model has several benefits, the main ones being its high resilience, greater efficiency, and capacity to handle big datasets. Fernando *et al.,* [7] used five distinct machine learning techniques, including the processes of GB, XGB, DT, RF, and ET, to discover and recognize intrusions from the network traffic information. Additionally, the authors have examined and evaluated the recommended approach's intrusion detection efficiency using a BoT-IoT dataset. Nevertheless, this work's precision falls short of expectations, which impairs the system's overall performance. Attou *et al.,* [20] deployed the Radial Basis Function Neural Network – Random Forest (RBFNN-RF) integrated classification technique to protect the cloud environment against intrusions. This project aims to reduce vulnerabilities and illegal data access by creating an automated intrusion detection system with a hybrid learning algorithm. The Intrusion Detection Systems (IDSs) [21] have become the most often used part of compliance and computing system security protocols for protecting cloud environments from various attacks and hazards. In order to keep up with the development of computer-related crimes, it employs numerous investigative techniques to find dangers, report criminal activity, and put preventative measures in action.

One of the main issues of today's technological age is network security. The weaknesses in network security have grown in importance over the past decade due to the internet's rapid expansion and widespread use. The purpose of an IDS is to detect abnormal attacks and unwanted access to protected networks. Numerous studies on IDS have been carried out in recent years [6,22]. IDS is able to execute an expert security analysis, identify and stop detrimental assaults on the network, and maintain regular operation throughout any malicious epidemic. These days, deep learning and sophisticated machine learning techniques are used to build IDS approaches. Within the

specialized field of artificial intelligence known as machine learning, information is obtained from training data using previously determined facts [23,24]. A science known as "machine learning" enables machines to acquire information without becoming deliberately programmed. For instance, the Decision Tree (DT), Naïve Bayes (NB), Artificial Neural Network (ANN), K-Nearest Neighbour (KNN), Fuzzy Logic (FL), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM) are the most frequently used artificial intelligence methodologies in the field of IoT security.

Al-Hadhrami *et al.,* [25] developed a real time attack detection framework for boosting security of IoT networks. The authors set out to look into the available datasets and how they might be used in IoT settings. Next, they talk about a methodology for gathering real-time data that may be used to create a dataset for testing and evaluating IDS. The primary benefit of the suggested dataset is that it includes characteristics specifically created for the 6LoWPAN/RPL network, which is probably the most used protocol in the IoT networks. Gyamfi *et al.,* [26] presented a detailed literature review to examine several approaches related to edge computing and machine learning for IoT security. By moving sophisticated computing operations from IoT devices to the edge, the multi-access edge computing (MEC) paradigm has arisen to alleviate these limitations. The majority of associated works currently in existence concentrate on identifying the best security ways to safeguard IoT devices. Additionally, a comparative examination of the assessment criteria, deployment methodologies, and datasets that are publicly available that were used in the IDS design is conducted in the current investigation. Geetha *et al.,* [27] established a novel principal component analysis model for cloud environment security that combines a deep learning method with a fisher kernel basis. PCA is used in this case to reduce dimensionality, and the GWO approach is used to select the best features. Additionally, using the selected features, the hybrid deep learning methodology is used to separate the normal and intrusion data from the input.

A recent assessment of the literature led to the conclusion that most intelligence algorithms have a number of complex issues, including a high false prediction rate, difficulty classifying threats, high processing costs, and system load. Thus, the suggested research project's objective is to enhance cloud-IoT system security by implementing a state-of-the-art security algorithm.

3. Proposed Methodology

This section provides the complete explanation for the proposed security model used to safeguard cloud-IoT system from modern cyber-threats. The main contribution of this paper is to develop a novel framework known as, Augmented Stacking Polynomial Optimized Tool (ASPOT) for increasing the security of cloud-IoT networks. Typically, it has been demonstrated that IDS is one of the effective and promising methods for maximizing the security of cloud-IoT systems. IDS has shown to be one of the most effective and promising strategies. Through the monitoring of system traffic data, it finds observed threats and malicious activity, and when such dangers are found, alarms are sent out. In the proposed system, the novel algorithms are implemented for analysing and detecting intrusions from the network traffic data. The flow of the proposed ASPOD system is shown in Figure 1, which encompasses the following stages of operations:

i. Data collection from cloud-IoT networks
ii. Deep Augmented Preprocessing Model (DAPM) for data normalization
iii. Binary Sand Cat Swarm Optimization (BSCSO) for feature selection
iv. Deep Stacking Polynomial Learning Network (DSPLN) for intrusion detection and classification

The current cyber-attack datasets that are accessible from open-source websites were utilized in this study for system analysis and implementation. Following the acquisition of data, DAPM is used to execute cleaning and normalizing procedures. This includes performing transformation, normalization, and augmentation activities. This methodology successfully improves data quality, which helps to achieve better intrusion detection performance. To guarantee an accurate classification, the dimensionality of the dataset is minimized by applying the BSCSO algorithm to extract the required features from the normalized data. Additionally, the last step uses the DSPLN methodology to classify the normal and intrusion data with a low false prediction rate. The suggested ASPOD model's intrusion detection results are significantly improved in the proposed framework by utilizing the proposed DAPM, BSCSO, and DSPLN techniques. Reduced computing costs, high precision, scalability, quick detection rates, identification of network risks in cloud-IoT contexts, and comparatively low false positive and false negative rates are the main benefits of utilizing the suggested approach.

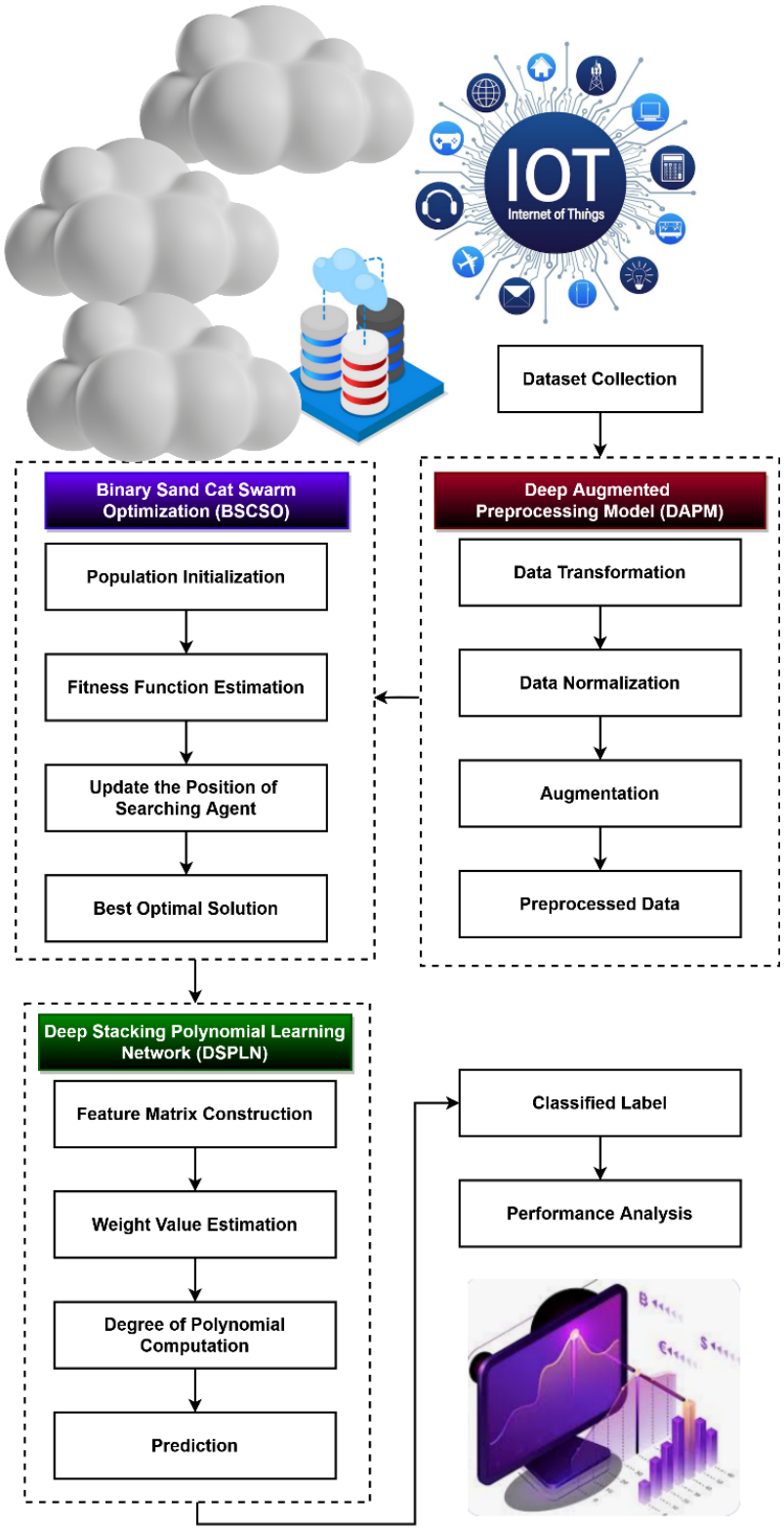

Fig. 1. Flow of the proposed ASPOT security model

3.1 DAPM for Data Preprocessing

The initial stage of machine learning is called "data pre-processing," in which the data gets altered or encoded to put it in a format that allows the computer to understand or parse it fast. In other words, it could also mean that the predictive method is able to quickly analyse the data's attributes.

The most significant factor influencing a supervised machine learning algorithm's ability to generalize is data preprocessing. The amount of data used for training increases dramatically with respect to the dimension of the input space. A prediction states that preprocessing can account for 50% to 80% of the total time required for identification, demonstrating the significance of preprocessing in model construction. Enhancing the quality of the data is also essential for enhanced efficiency. Finding inaccurate or noisy data and fixing it or eliminating it from the collection of information is the procedure of data cleaning. It generally functions to detect and replace any noisy, erroneous, incomplete, or superfluous records or information. Although the methods employed change based on the model's requirements, the fundamental actions taken during this process are:

i. Elimination of redundant or unnecessary information.
ii. Setting structural mistakes
iii. Missing values handling

The proposed DAPM algorithm is employed to broaden the data and increase the samples that account for a smaller percentage of the initial training set, balancing the proportion of the majority and minority samples. This is because the overall amount of attacking samples in the given dataset is significantly higher than that in the set used for training, and only a small percentage of these samples are in the training set. As a result, the trained model finds it difficult to distinguish these samples. This can help to some extent with the imbalance issue in the data from the network and improve the model's capacity for extrapolation. A huge gap between various dimensional data points for features within the dataset might result in issues like weak model training and negligible accuracy rise. To address this issue, the min-max function is used to map the data within a value range of (0, 1) in the following way:

$$\mathrm{d}' = \frac{\mathrm{d}-\mathrm{d}_{\min}}{\mathrm{d}_{\max}-\mathrm{d}_{\min}} \quad (1)$$

Where, $\mathrm{d}_{\max}$ and $\mathrm{d}_{\min}$ are the maximum and minimum values of the data.

3.2 BSCSO for Feature Selection

The massive amount of traffic generated by the cloud-IoT network significantly down the intrusion detection operation. It's crucial to select only the pertinent data because, for detecting reasons, the data frequently contains some redundant and unimportant data. Because feature selection may efficiently discover a subset of the most appropriate features in the dataset according to specific criteria. Also, it is a crucial component of any network detection system for intrusions that aims to decrease processing time and boost system reliability. The efficiency of the system is not adversely affected by eliminating those superfluous features. Additionally, utilizing each attribute makes the system more complex and less accurate. Therefore, the objective is to choose the most effective subset of attributes that are pertinent to the intrusion detection operation. Consequently, boosting the efficiency of information mining algorithms is the primary focus of the selection of features. By cutting out features that are superfluous or redundant, we can improve system accuracy and generate classification models with greater speed, besides numerous other benefits. In the proposed ASPOT model, the novel and unique optimization technique, named as, BSCSO has been employed to optimally pick the essential features for training the classifier’s features. The sand cat has an extremely unique way of finding food and hunting. These animals' remarkable knack to discover prey is based on their capacity to find prey on the earth or beneath. They can therefore

locate their prey with great speed. In order to identify the best solution, the swarm optimization algorithm (SCSO) mimicked this characteristic. Initializing the population is the first stage, just like in other meta-heuristic algorithms. The lower and upper bounds of the problem have been employed to randomly generate the search space. A search agent solution to a predetermined optimization problem is shown by each segment of the search field.

When compared to the other meta-heuristic models, the proposed BSCSO has the primary advantages of simple to implement, low searching complexity, and high efficiency. Meta-heuristic algorithms aim to discover a solution that is as close to optimum as feasible. In this manner, a cost function or fitness function is established for the feature selection problem in order to assess the resultant solution. The goal of the solution is guided by the meta-heuristic algorithm, which is based on the problem objective. Up until the final iteration, the fitness of each solution dictates the next iteration. The most ideal option had been identified in the last iteration, which the user can opt to utilize. At this point, every mechanism in a meta-heuristic algorithm finds the best solution, which is usually determined by the hunting mechanism. There is a unique principle of operation for the SCSO algorithm. Finding the most effective solution involves looking for prey after initiation. This makes use of the sand cat's capacity for noise with low frequencies result. In this technique, the set of population is initialized at the beginning of optimization, and the fitness function is calculated according to the objective function. Until reaching the maximum number of iterations, the random angle is selected according to the Roulette wheel selection method. Then, the position of searching is updated as represented in the following equation:

$$\vec{\mathrm{B}}(\mathrm{k}+1)=\begin{cases}\overrightarrow{\rho_{\mathrm{b}}}(\mathrm{k})-\overrightarrow{\rho_{\mathrm{r}}}\times\cos(\theta)\times\vec{\omega} & |\mathrm{G}|\leq 1;\\ \vec{\omega}\times\left(\overrightarrow{\rho_{\mathrm{bp}}}(\mathrm{k})-\mathrm{M}(0{,}1)\right)\times\overrightarrow{\rho_{\mathrm{p}}}(\mathrm{k}) & |\mathrm{G}|>1\end{cases} \tag{2}$$

$$\vec{\omega}=\overrightarrow{\omega_q}\times M(0{,}1) \tag{3}$$

Where, $\vec{B}(k+1)$ is the updated position of searching agent, $\vec{\omega}$ denotes the sensitivity range, M is the $\overrightarrow{\rho_b}$ represents the best position, $\overrightarrow{\rho_p}$ indicates the current position, and θ is the random angle. Consequently, the v-shaped transfer function is applied for integrating the binary algorithm with SCSO technique as represented in the following equation:

$$\mathcal{V}\left(b_i^n(k)\right)=\frac{2}{\pi}arctan\left(\frac{\pi}{2}b_i^n(k)\right) \tag{4}$$

$$b_i^n(k)=\begin{cases}(b_i^n)^{-1} & if\ M<\mathcal{V}\left(b_i^n(k)\right)\\ (b_i^n) & Otherwise\end{cases} \tag{5}$$

This model states that in order to extract the best necessary features from the dataset, the best optimal solution is found and used.

Algorithm 1 – BSCSO for Feature Selection

Input: Pre-processed dataset P_D;
Output: Selected features S_f;
Begin
Step 1: Initialize the optimization input parameters, maximum number of iterations, current iteration, and set of population;

Step 2: Compute the fitness function according to the objective function;
Step 3: Initialize the parameters sensitivity range $\vec{\omega}$, current position $\overrightarrow{\rho_p}$, and best position $\overrightarrow{\rho_b}$;
Step 4: While ($itr \leq mx_{itr}$)
For each searching agent in the field

Obtain a random angle according to the roulette wheel selection method;
If $(abs\ (Q) \leq 1)$
Update the position of searching agent as shown in Eq. (2);

$$x = \overrightarrow{\rho_b}(k) - \overrightarrow{\rho_r} \times cos\ (\theta) \times \vec{\omega} \tag{6}$$

Else
Update the position of searching agent as shown in Eq. (2);

$$y = \vec{\omega} \times (\overrightarrow{\rho_b}(k) - M(0,1) \times \overrightarrow{\rho_p}) \tag{7}$$

End

Compute the v-transfer function $\mathcal{V}\left(b_i^n(k)\right)$ as shown in Eq. (4);
If $(M \leq \mathcal{V}\left(b_i^n(k)\right))$
Update the position $b_i^n(k)$ of searching agent as shown in Eq. (5);
Else
Update the position $b_i^n(k)$ of searching agent as shown in Eq. (5);
End;
End for;
$k = k + 1$;
End;
Step 5: Return the best optimal features $S_f = b_i^n(k)$;

3.3 DSPLN for Intrusion Classification

In the classification process, the normal and intrusion data is categorized with the use of DSPLN technique, which uses the best optimal features obtained from the previous stage as the input. In the conventional studies, a variety of classification approaches are implemented for attack identification and classification. Here, the unique reasons of adopting the proposed DSPLN methodology are increased prediction rate, low false rate, reduced time for execution, and simple to implement. A deep hierarchy can be created in DSPLN by stacking many basic deep probability networks on top of one another. The output of each node in this unique supervised deep neural network technique is a quadratic function derived from its components, and it includes effective layer-by-layer learning. A DSPLN classifier receives the output model at the highest level beyond that. The DSPLN training time is reduced by removing unnecessary characteristics by the selection of optimal features. Furthermore, the best way to construct deep networks in DSPLN enables learning of data representation on small bound instances. The approach begins with a straightforward network that might have a sizable bias but aren't going often over-fit, and as the system gets more complex, we are able to decrease the degree of bias while boosting the variance. As a result, the intrinsic curves of the solutions that govern the bias-variance compromise might be developed with this method of learning. In order to accumulate an independent collection of vectors to generate the bias, Singular Vector Decomposition (SVD) is used to define the degree-1 polynomial function (linear)

throughout the dataset used for training. This serves as the initial phase in the development of the layers in deep probability network. All values produced by the linear functions used on the training cases are extended in the first layer's outputs. The basis for degree-2 polynomials can then be obtained using the same idea. This indicates that the array of values bought by nodes in the first layer and the outcomes of the outputs of every two elements in the first layer are augmented to encompass the matrix of values resulting from any degree-2 polynomial. This technique predicts the resultant label based on this operation and uses it to accurately classify the normal and intrusion data.

4. Results and Discussion

This section uses a number of metrics and open-source datasets to evaluate and assess the performance of the suggested ASPOT approach. Some of the most well-known and recent cyber-attack datasets, including UNSW-NB 15 [28], ToN-IoT [29], and CSECIC-2018, have been used for testing and validation in this work. Tables 1 and 2 provide the dataset descriptions for ToN-IoT and UNSW-NB 15, respectively.

Table 1
Dataset description for ToN-IoT

Classes of attacks	No of samples
Normal	3,00,000
Backdoor	20,000
DoS	20,000
DDoS	20,000
Injection	20,000
MITM	1,043
Password	20,000
Ransomware	20,000
Scanning	20,000
XSS	20,000

Table 2
Dataset description for UNSW-NB

Classes of attacks	No of samples for training	No of samples for testing
Normal	56,000	37,000
Analysis	2,000	677
Backdoor	1,746	583
DoS	12,264	4089
Exploits	33,393	11,132
Fuzzers	18,184	6,062
Generic	40,000	18,871
Reconnaissance	10,491	3,496
Shellcode	1,133	378
Worms	130	44

One of the most crucial things to be concerned about is selecting an appropriate performance metric. The most popular metric for illustrating the system's effectiveness is its accuracy. The accuracy of the classifier can be defined as the number of cases correctly identified divided by the total number of cases. The parameter of accuracy is calculated by using the following equation:

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \quad (8)$$

Precision, sometimes referred to as positive predictive value, is the likelihood that the model would properly identify any event, which is computed as shown in the following model:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{9}$$

The "true positive rate" refers to the percentage of test outcomes that the model properly detected. It is often referred to as the "detection rate" or the "recall rate." Most likely, another name for it is sensitivity as computed by using the following equation:

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FP}} \tag{10}$$

Subsequently, the F-score is the harmonic mean of recall and accuracy, which is calculated as shown in below:

$$\text{F1} - \text{score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{\text{Precision+Recall}} \tag{11}$$

Where, TP – true positive, TN – true negative, FP – false positive, and FN – false negative. The UNSW-NB 15 dataset is used in Figure 2 and Table 3 to compare the overall performance of the suggested ASPOT methodology.

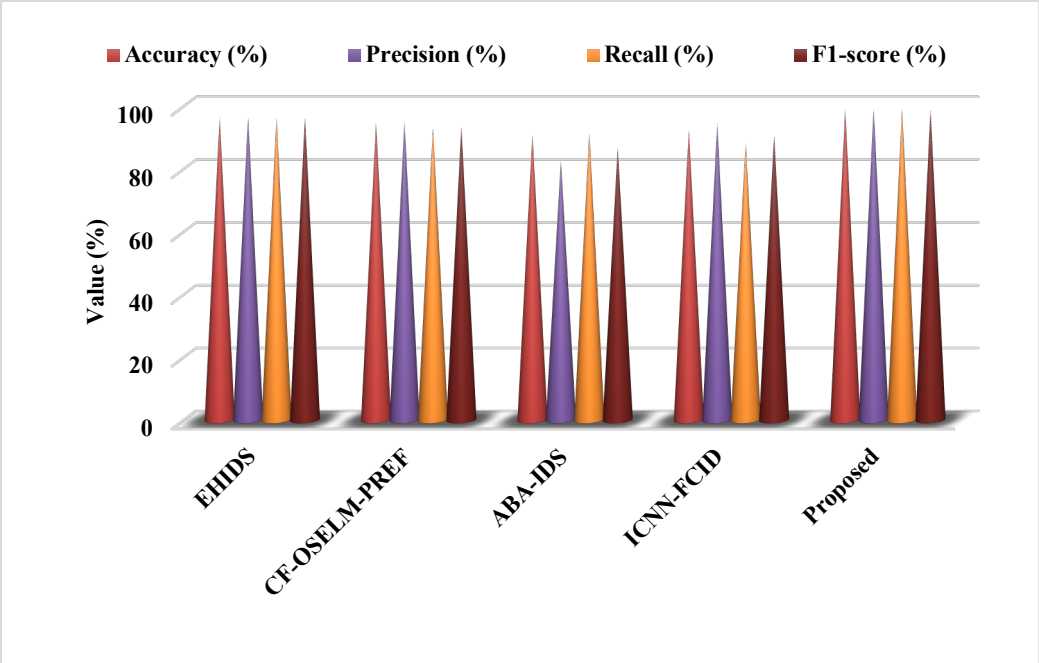

Fig. 2. Overall performance comparative study using UNSW-NB 15 dataset

Table 3
Comparative analysis using UNSW-NB 15 dataset

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
EHIDS	96.47	96.53	96.47	96.47
CF-OSELM-PREF	94.70	94.94	93.12	93.54
ABA-IDS	90.88	82.50	91.37	86.83
ICNN-FCID	92.38	94.44	88.41	90.66
Proposed	99	99.2	99.1	99

Likewise, as illustrated in Figure 3 and Table 4, the same performance metrics are calculated for the suggested and standard approaches. The predicted outputs show that the suggested ASPOT methodology outperforms all other approaches with excellent performance outcomes. As one of the main factors contributing to the enhanced results in the planned work is the implementation of BSCSO.

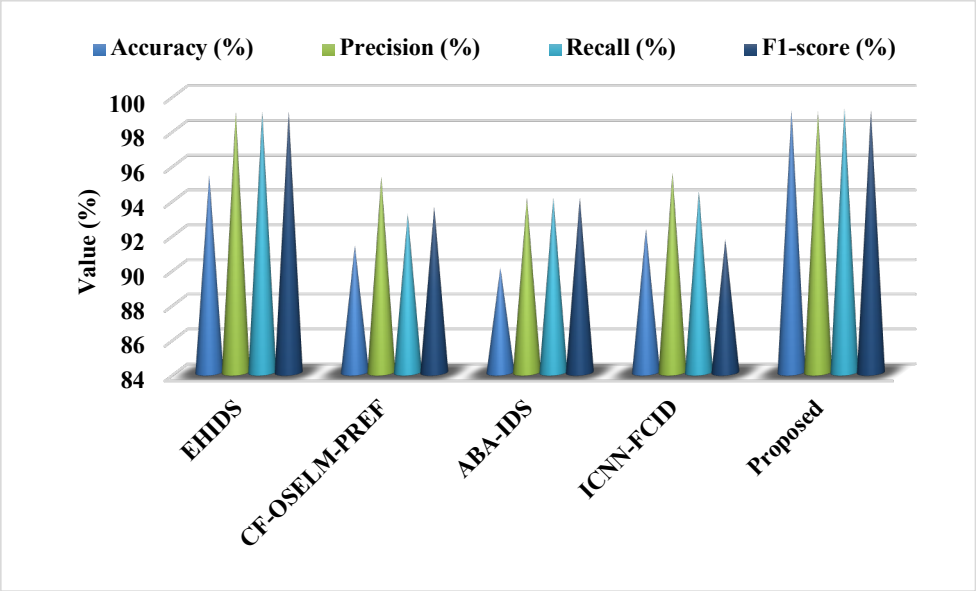

Fig. 3. Overall performance comparative study using ToN-IoT dataset

Table 4
Comparative analysis using ToN-IoT dataset

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
EHIDS	95.36	99.02	99.01	99.02
CF-OSELM-PREF	91.31	95.27	93.12	93.54
ABA-IDS	90.03	94.05	94.05	94.05
ICNN-FCID	92.26	95.51	94.43	91.66
Proposed	99.1	99	99.2	99.1

Furthermore, using the UNSW-NB 15 and ToN-IoT datasets, Table 5 and Figure 4 compare the execution times of the suggested [30] and standard security techniques. According to the evaluation, the suggested ASPOT methodology's execution time is shortened to 1.2 seconds for the UNSW-NB 15 data and 1.542 seconds for the ToN-IoT dataset. The suggested ASPOT framework's execution time has been significantly shortened as a result of the use of the BSCSO and DSPLN approaches, since decreased dimensionality is a key factor in reducing classifier time.

Table 5
Execution time analysis

Methods	UNSW-NB 15	ToN-IoT
EHIDS	2.461	1.832
CF-OSELM-PREF	2.949	3.242
ABA-IDS	2.951	2.946
ICNN-FCID	3.309	2.911
Proposed	1.269	1.542

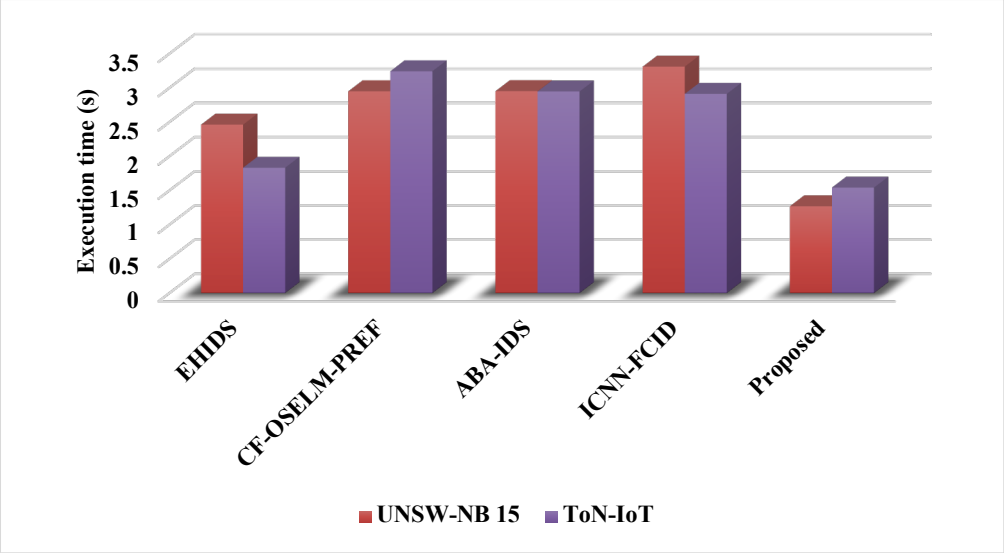

Fig. 4. Comparison based on execution time

The accuracy, precision, recall, and f1-score values of the suggested and standard ASPOT security approaches are contrasted with regard to different numbers of samples in the UNSW-NB 15 dataset, respectively, in Figure 5 to Figure 8.

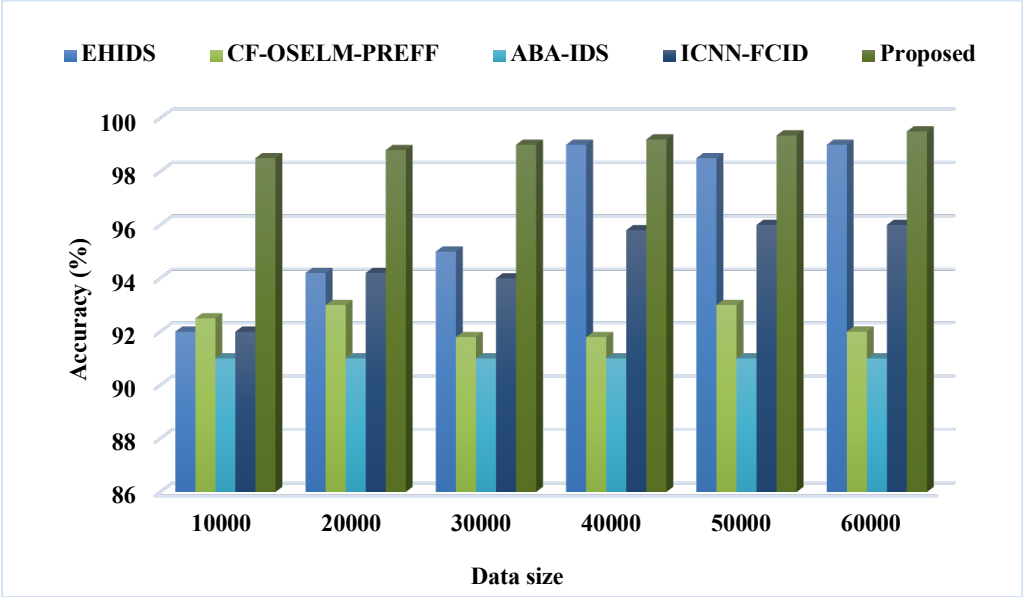

Fig. 5. Accuracy analysis with respect to varying number of samples in UNSW-NB 15 dataset

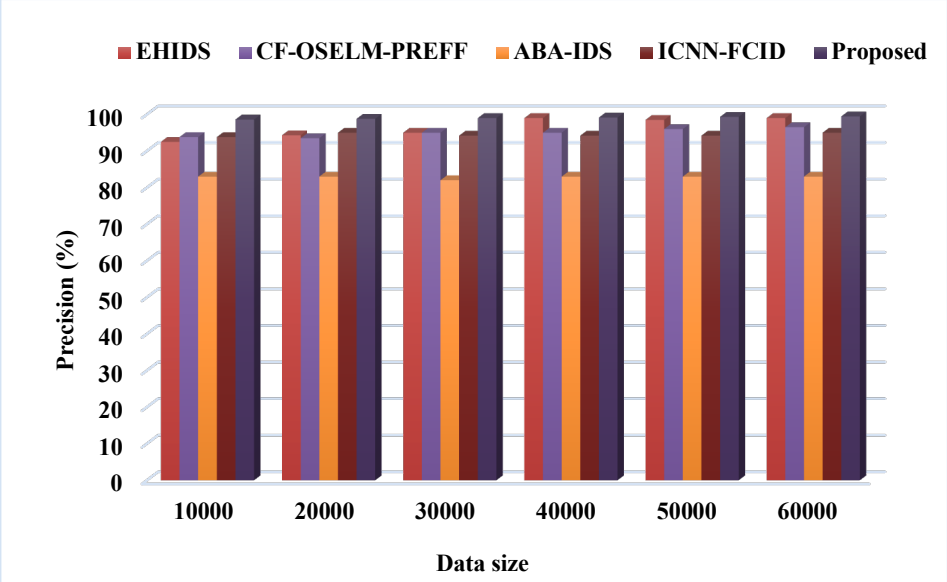

Fig. 6. Precision analysis with respect to varying number of samples in UNSW-NB 15 dataset

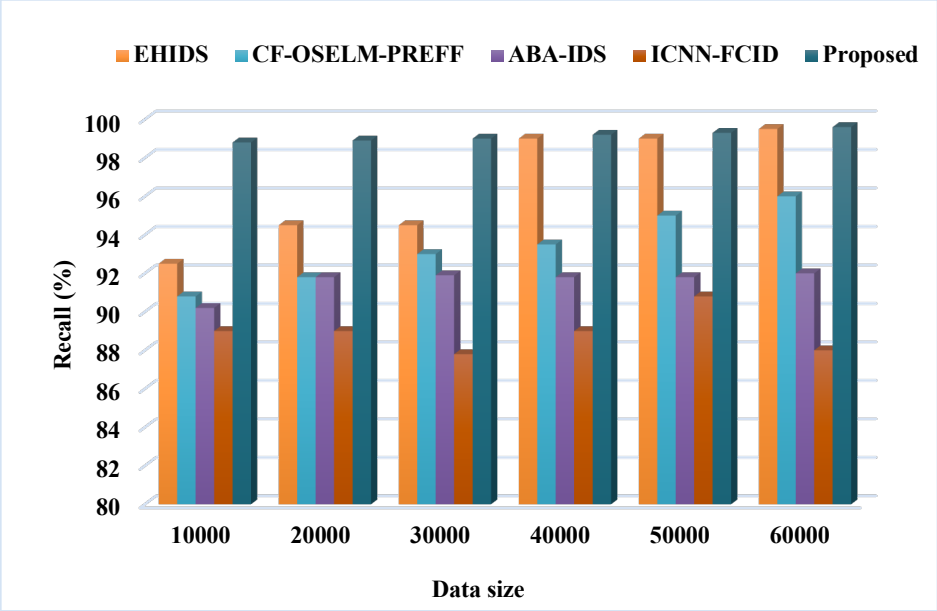

Fig. 7. Recall analysis with respect to varying number of samples in UNSW-NB 15 dataset

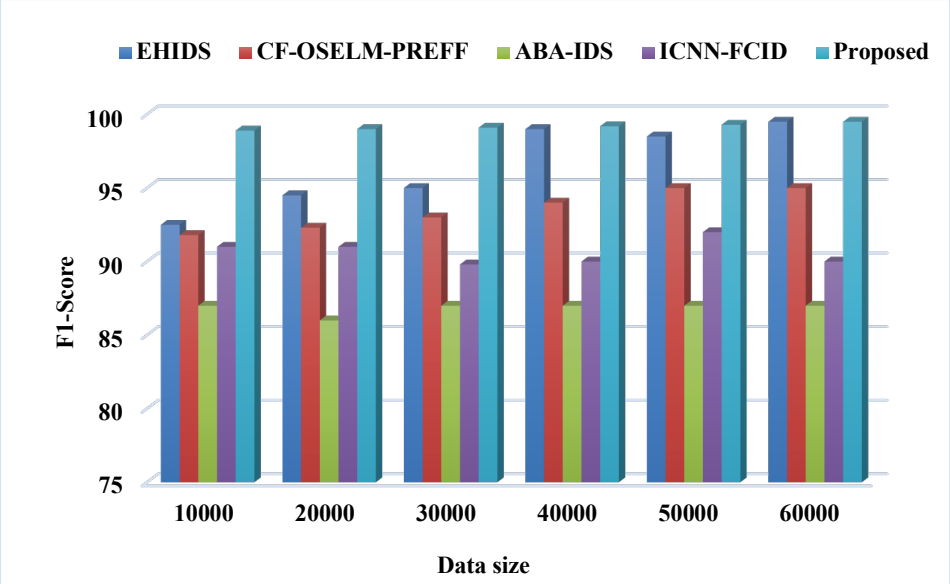

Fig. 8. F1-score analysis with respect to varying number of samples in UNSW-NB 15 dataset

Their corresponding values are tabulated in Table 6 to Table 9.

Table 6
Comparison based on accuracy with respect to varying data samples in UNSW-NB15 dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	92	92.5	91	92	98.5
20000	94.2	93	91	94.2	98.8
30000	95	91.8	91	94	99
40000	99	91.8	91	95.8	99.2
50000	98.5	93	91	96	99.35
60000	99	92	91	96	99.5

Table 7
Comparison based on precision with respect to varying data samples in UNSW-NB15 dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	92.5	93.8	83	93.8	98.6
20000	94.3	93.5	83	95	98.8
30000	95	95	82	94.2	99
40000	99	95	83	94.2	99.15
50000	98.5	96	83	94.2	99.33
60000	99	96.5	83	95	99.45

This comparison analysis shows that the suggested ASPOT methodology outperforms all other traditional methods with higher recall, accuracy, precision, and f1-score values. Improved performance outcomes in the suggested security framework are usually the result of integrating new approaches like DAPM, BSCSO, and DSPLN.

Table 8
Comparison based on recall with respect to varying data samples in UNSW-NB15 dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	92.5	90.8	90.2	89	98.8
20000	94.5	91.8	91.8	89	98.9
30000	94.5	93	91.9	87.8	99
40000	99	93.5	91.8	89	99.2
50000	99	95	91.8	90.8	99.3
60000	99.5	96	92	88	99.6

Table 9
Comparison based on f1-score with respect to varying data samples in UNSW-NB15 dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	92.5	91.8	87	91	98.9
20000	94.5	92.3	86	91	99
30000	95	93	87	89.8	99.1
40000	99	94	87	90	99.2
50000	98.5	95	87	92	99.3
60000	99.5	95	87	90	99.5

Similarly, using the ToN-IoT dataset, Figure 9 to Figure 12 validate and compare the same performance measures.

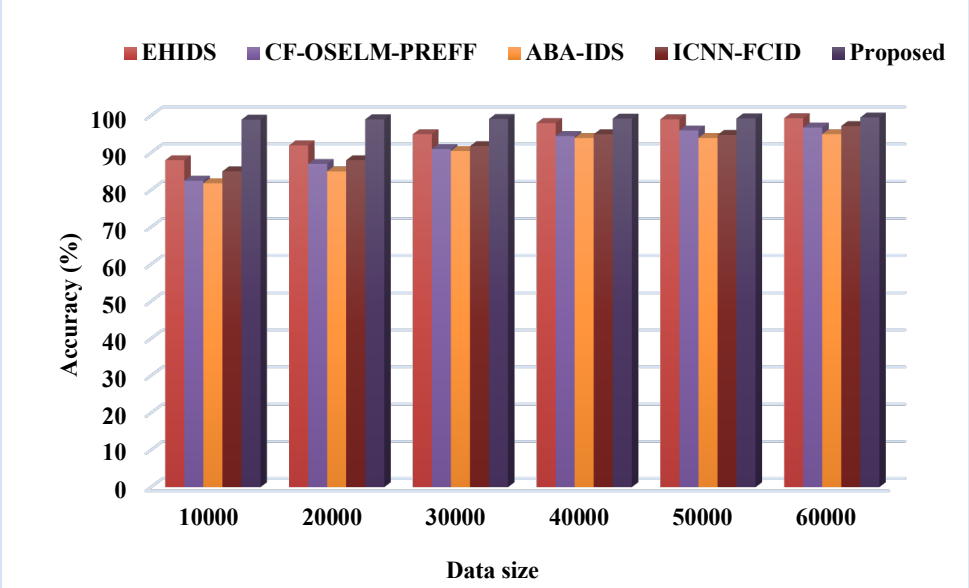

Fig. 9. Accuracy analysis with respect to varying number of samples in ToN-IoT dataset

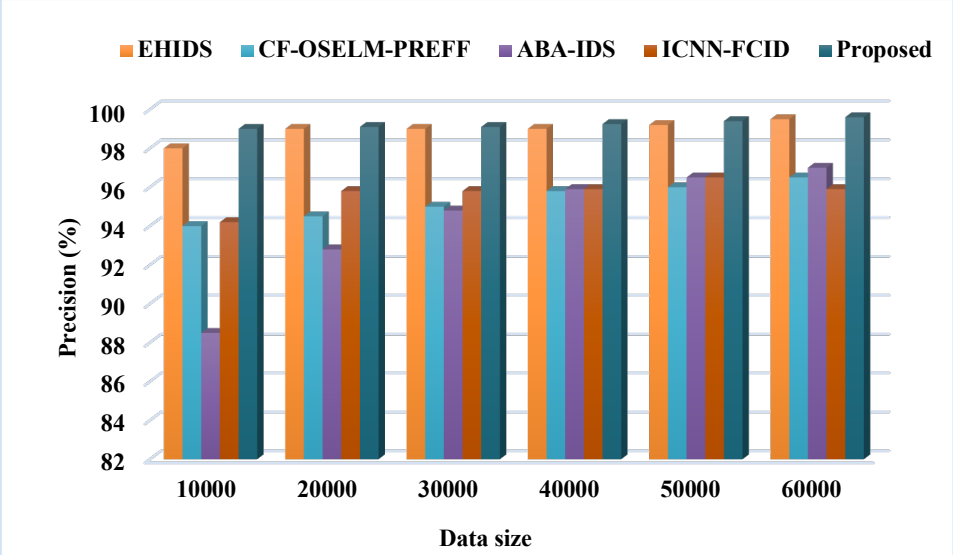

Fig. 10. Precision analysis with respect to varying number of samples in ToN-IoT dataset

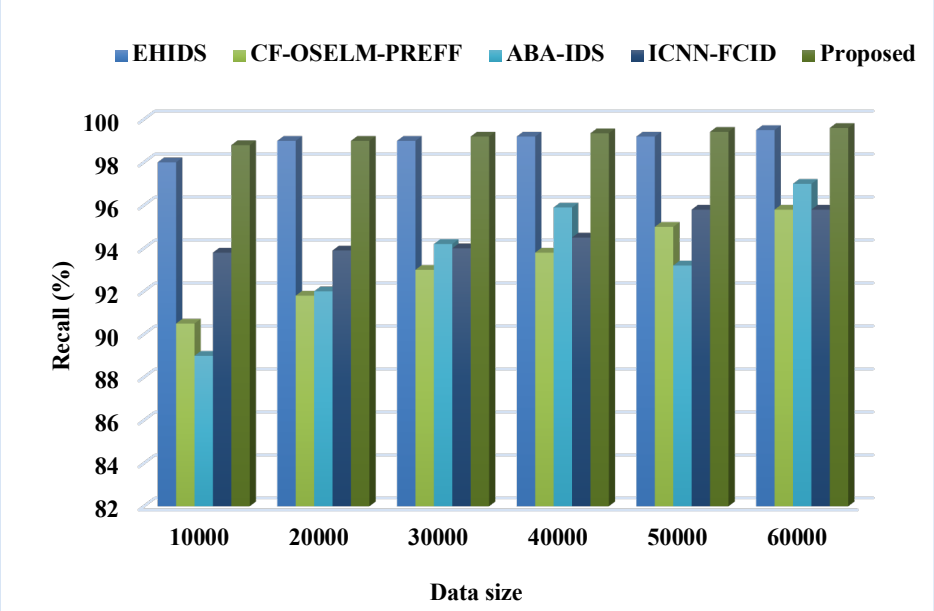

Fig. 11. Recall analysis with respect to varying number of samples in ToN-IoT dataset

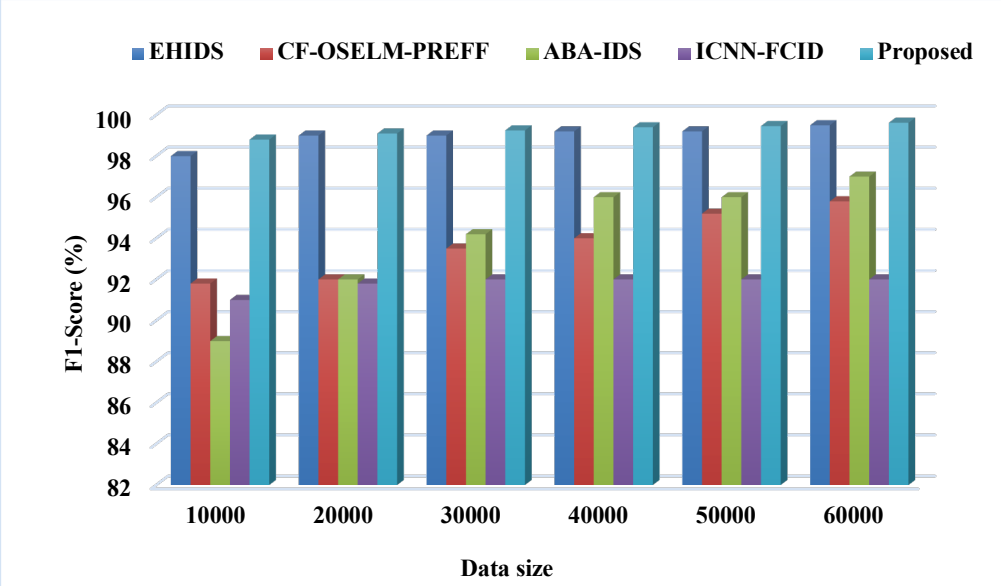

Fig. 12. F1-score analysis with respect to varying number of samples in ToN-IoT dataset

The relevant values are reported in Tables 10 to 13, respectively. This estimate makes it clear that, in comparison to the other security approaches, the ASPOT methodology yields better performance results.

Table 10
Comparison based on accuracy with respect to varying data samples in ToN-IoT dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	88	82.5	81.8	85	98.9
20000	92	87	85	88	99
30000	95	91	90.5	91.8	99.1
40000	98	94.5	94	95	99.2
50000	99	96	94	94.8	99.25
60000	99.3	96.8	95	97.2	99.5

Table 11
Comparison based on precision with respect to varying data samples in ToN-IoT dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	98	94	88.5	94.2	99
20000	99	94.5	92.8	95.8	99.1
30000	99	95	94.8	95.8	99.1
40000	99	95.8	95.9	95.9	99.25
50000	99.2	96	96.5	96.5	99.4
60000	99.5	96.5	97	95.9	99.6

Table 12
Comparison based on recall with respect to varying data samples in ToN-IoT dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	98	90.5	89	93.8	98.8
20000	99	91.8	92	93.9	99
30000	99	93	94.2	94	99.2
40000	99.2	93.8	95.9	94.5	99.35
50000	99.2	95	93.2	95.8	99.42
60000	99.5	95.8	97	95.8	99.6

Table 13
Comparison based on f1-score with respect to varying data samples in ToN-IoT dataset

Data samples	EHIDS	CF-OSELM-PREFF	ABA-IDS	ICNN-FCID	Proposed
10000	98	91.8	89	91	98.8
20000	99	92	92	91.8	99.1
30000	99	93.5	94.2	92	99.25
40000	99.2	94	96	92	99.4
50000	99.2	95.2	96	92	99.46
60000	99.5	95.8	97	92	99.62

4. Conclusion

This paper's primary goal is to create the innovative security framework known as ASPOT in order to defend cloud-IoT against cyberattacks. This study used open-source websites to obtain the current cyber-attack datasets for system analysis and implementation. DAPM is used to carry out cleaning and normalization processes after data collection. This involves carrying out tasks related to augmentation, normalization, and transformation. By successfully enhancing data quality, this methodology contributes to improved intrusion detection performance. By using the BSCSO technique to extract the necessary features from the normalized data, the dimensionality of the dataset is reduced, ensuring a correct classification. Furthermore, the final stage classifies the normal and intrusion data with a low false prediction rate using the DSPLN methodology. Using the proposed DAPM, BSCSO, and DSPLN approaches greatly improves the intrusion detection results of the provided ASPOD model in the proposed framework. The key advantages of using the proposed approach include lower computational costs, high precision, scalability, fast detection rates, network risk identification in cloud-IoT settings, and relatively low false positive and false negative rates. This work has leveraged a number of popular and recent cyber-attack datasets for testing and validation, such as UNSW-NB 15, ToN-IoT, and CSECIC-2018. The comparison analysis demonstrates that, with higher recall, accuracy, precision, and f1-score values than any other traditional method, the suggested ASPOT methodology performs better. Integrating novel techniques like DAPM, BSCSO, and DSPLN typically leads to better performance outcomes in the recommended security framework.

Acknowledgement
This research was not funded by any grant.

References

[1] Syed, Naeem Firdous, Mengmeng Ge, and Zubair Baig. "Fog-cloud based intrusion detection system using Recurrent Neural Networks and feature selection for IoT networks." *Computer Networks* 225 (2023): 109662. https://doi.org/10.1016/j.comnet.2023.109662

[2] Yi, Lizhi, Mei Yin, and Mehdi Darbandi. "A deep and systematic review of the intrusion detection systems in the fog environment." *Transactions on Emerging Telecommunications Technologies* 34, no. 1 (2023): e4632. https://doi.org/10.1002/ett.4632

[3] Ahmad, Shahnawaz, Shabana Mehfuz, and Javed Beg. "An efficient and secure key management with the extended convolutional neural network for intrusion detection in cloud storage." *Concurrency and Computation: Practice and Experience* 35, no. 23 (2023): e7806. https://doi.org/10.1002/cpe.7806

[4] Attou, Hanaa, Azidine Guezzaz, Said Benkirane, Mourade Azrour, and Yousef Farhaoui. "Cloud-Based Intrusion Detection Approach Using Machine Learning Techniques." *Big Data Mining and Analytics* 6, no. 3 (2023): 311-320. https://doi.org/10.26599/BDMA.2022.9020038

[5] Lazzarini, Riccardo, Huaglory Tianfield, and Vassilis Charissis. "A stacking ensemble of deep learning models for IoT intrusion detection." *Knowledge-Based Systems* 279 (2023): 110941. https://doi.org/10.1016/j.knosys.2023.110941

[6] Zhao, Guosheng, Yang Wang, and Jian Wang. "Lightweight Intrusion Detection Model of the Internet of Things with Hybrid Cloud-Fog Computing." *Security and Communication Networks* 2023 (2023). https://doi.org/10.1155/2023/7107663

[7] Fernando, Gutierrez-Portela, Arteaga-Arteaga Harold Brayan, Almenares Mendoza Florina, Calderón-Benavides Liliana, Acosta-Mesa Héctor-Gabriel, and Tabares-Soto Reinel. "Enhancing Intrusion Detection in IoT Communications Through ML Model Generalization With a New Dataset (IDSAI)." *IEEE Access* (2023). https://doi.org/10.1109/ACCESS.2023.3292267

[8] Binbusayyis, Adel. "Hybrid VGG19 and 2D-CNN for intrusion detection in the FOG-cloud environment." *Expert Systems with Applications* 238 (2024): 121758. https://doi.org/10.1016/j.eswa.2023.121758

[9] Chaganti, Rajasekhar, Wael Suliman, Vinayakumar Ravi, and Amit Dua. "Deep learning approach for SDN-enabled intrusion detection system in IoT networks." *Information* 14, no. 1 (2023): 41. https://doi.org/10.3390/info14010041

[10] Gopi, R., R. Sheeba, K. Anguraj, T. Chelladurai, Haya Mesfer Alshahrani, Nadhem Nemri, and Tarek Lamoudan. "Intelligent Intrusion Detection System for Industrial Internet of Things Environment." *Computer Systems Science & Engineering* 44, no. 2 (2023). https://doi.org/10.32604/csse.2023.025216

[11] Soliman, Sahar, Wed Oudah, and Ahamed Aljuhani. "Deep learning-based intrusion detection approach for securing industrial Internet of Things." *Alexandria Engineering Journal* 81 (2023): 371-383. https://doi.org/10.1016/j.aej.2023.09.023

[12] Vaiyapuri, Thavavel, Shabbab Algamdi, Rajan John, Zohra Sbai, Munira Al-Helal, Ahmed Alkhayyat, and Deepak Gupta. "Metaheuristics with federated learning enabled intrusion detection system in Internet of Things environment." *Expert Systems* 40, no. 5 (2023): e13138. https://doi.org/10.1111/exsy.13138

[13] Kethineni, Keerthi, and G. Pradeepini. "Intrusion detection in internet of things-based smart farming using hybrid deep learning framework." *Cluster Computing* (2023): 1-14. https://doi.org/10.1007/s10586-023-04052-4

[14] Raj, Meghana G., and Santosh Kumar Pani. "Hybrid feature selection and BWTDO enabled DeepCNN-TL for intrusion detection in fuzzy cloud computing." *Soft Computing* (2023): 1-20. https://doi.org/10.1007/s00500-023-08573-3

[15] Vashishtha, Lalit Kumar, Akhil Pratap Singh, and Kakali Chatterjee. "HIDM: A hybrid intrusion detection model for cloud based systems." *Wireless Personal Communications* 128, no. 4 (2023): 2637-2666. https://doi.org/10.1007/s11277-022-10063-y

[16] Dina, Ayesha S., A. B. Siddique, and D. Manivannan. "A deep learning approach for intrusion detection in Internet of Things using focal loss function." *Internet of Things* 22 (2023): 100699. https://doi.org/10.1016/j.iot.2023.100699

[17] Mohy-Eddine, Mouaad, Azidine Guezzaz, Said Benkirane, Mourade Azrour, and Yousef Farhaoui. "An Ensemble Learning Based Intrusion Detection Model for Industrial IoT Security." *Big Data Mining and Analytics* 6, no. 3 (2023): 273-287. https://doi.org/10.26599/BDMA.2022.9020032

[18] Elnakib, Omar, Eman Shaaban, Mohamed Mahmoud, and Karim Emara. "EIDM: deep learning model for IoT intrusion detection systems." *The Journal of Supercomputing* (2023): 1-21. https://doi.org/10.1007/s11227-023-05197-0

[19] Abd Elaziz, Mohamed, Mohammed AA Al-qaness, Abdelghani Dahou, Rehab Ali Ibrahim, and Ahmed A. Abd El-Latif. "Intrusion detection approach for cloud and IoT environments using deep learning and Capuchin Search

Algorithm." *Advances in Engineering Software* 176 (2023): 103402. https://doi.org/10.1016/j.advengsoft.2022.103402

[20] Attou, Hanaa, Mouaad Mohy-eddine, Azidine Guezzaz, Said Benkirane, Mourade Azrour, Abdulatif Alabdultif, and Naif Almusallam. "Towards an intelligent intrusion detection system to detect malicious activities in cloud computing." *Applied Sciences* 13, no. 17 (2023): 9588. https://doi.org/10.3390/app13179588

[21] Roy, Souradip, Juan Li, and Yan Bai. "A two-layer fog-cloud intrusion detection model for IoT networks." *Internet of Things* 19 (2022): 100557. https://doi.org/10.1016/j.iot.2022.100557

[22] Telikani, Akbar, Jun Shen, Jie Yang, and Peng Wang. "Industrial IoT intrusion detection via evolutionary cost-sensitive learning and fog computing." *IEEE Internet of Things Journal* 9, no. 22 (2022): 23260-23271. https://doi.org/10.1109/JIOT.2022.3188224

[23] Ramana, T. V., M. Thirunavukkarasan, Amin Salih Mohammed, Ganesh Gopal Devarajan, and Senthil Murugan Nagarajan. "Ambient intelligence approach: Internet of Things based decision performance analysis for intrusion detection." *Computer Communications* 195 (2022): 315-322. https://doi.org/10.1016/j.comcom.2022.09.007

[24] Lilhore, Umesh Kumar, Poongodi Manoharan, Sarita Simaiya, Roobaea Alroobaea, Majed Alsafyani, Abdullah M. Baqasah, Surjeet Dalal, Ashish Sharma, and Kaamran Raahemifar. "HIDM: Hybrid Intrusion Detection Model for Industry 4.0 Networks Using an Optimized CNN-LSTM with Transfer Learning." *Sensors* 23, no. 18 (2023): 7856. https://doi.org/10.3390/s23187856

[25] Al-Hadhrami, Yahya, and Farookh Khadeer Hussain. "Real time dataset generation framework for intrusion detection systems in IoT." *Future Generation Computer Systems* 108 (2020): 414-423. https://doi.org/10.1016/j.future.2020.02.051

[26] Gyamfi, Eric, and Anca Jurcut. "Intrusion detection in internet of things systems: a review on design approaches leveraging multi-access edge computing, machine learning, and datasets." *Sensors* 22, no. 10 (2022): 3744. https://doi.org/10.3390/s22103744

[27] Geetha, T. V., and A. J. Deepa. "A FKPCA-GWO WDBiLSTM classifier for intrusion detection system in cloud environments." *Knowledge-Based Systems* 253 (2022): 109557. https://doi.org/10.1016/j.knosys.2022.109557

[28] Tahri, Rachid, Abdessamad Jarrar, Abdellatif Lasbahani, and Youssef Balouki. "A comparative study of Machine learning Algorithms on the UNSW-NB 15 Dataset." In *ITM Web of Conferences*, vol. 48, p. 03002. EDP Sciences, 2022. https://doi.org/10.1051/itmconf/20224803002

[29] Khanday, Shahbaz Ahmad, Hoor Fatima, and Nitin Rakesh. "Implementation of intrusion detection model for DDoS attacks in Lightweight IoT Networks." *Expert Systems with Applications* 215 (2023): 119330. https://doi.org/10.1016/j.eswa.2022.119330

[30] Mohamed, Doaa, and Osama Ismael. "Enhancement of an IoT hybrid intrusion detection system based on fog-to-cloud computing." *Journal of Cloud Computing* 12, no. 1 (2023): 1-13. https://doi.org/10.1186/s13677-023-00420-y

[31] Ahmad, Waqas, Aamir Rasool, Abdul Rehman Javed, Thar Baker, and Zunera Jalil. "Cyber security in IoT-based cloud computing: A comprehensive survey." *Electronics* 11, no. 1 (2021): 16. https://doi.org/10.3390/electronics11010016

[32] Zin, Muhamad Zulfikri Md, Raihana Md Saidi, Faridah Sappar, and Mohamad Asrol Arshad. "Multi-factor Authentication to Authorizing Access to an Application: A Conceptual Framework." (2019).

[33] Taib, Abidah Mat, and Nurul Nabila Khairu Azman Azman. "Experimental Analysis of Trojan Horse and Worm Attacks in Windows Environment."

[34] Rahman, Teh Faradilla Abdul, and Zamri Abu Bakar. "Exploring Users' Perspectives on Co-Curricular Registration System using TAM Model." *Journal of Advanced Research in Computing and Applications* 16, no. 1 (2019): 24-33.

[35] Othman, Intan Safina, Abdul Samad Shibgatullah, Abd Samad Hassan Basari, Zul Azri, and Muhammad Noh. "The awareness of security breach among IT users in Kolej PolyTech MARA, Batu Pahat."