ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20230130000295
Original scientific paper

Seagull Optimization-based Feature Selection with Optimal Extreme Learning Machine for Intrusion Detection in Fog Assisted WSN

Thiruppathi MUTHU, Vinoth Kumar KALIMUTHU*

Abstract: On the internet, various devices that are connected to the Internet of Things (IoT) and Wireless Sensor Networks (WSNs) share the resources that they have in accordance with their respective needs. The information gathered from these Internet of Things devices was preserved in the cloud. The problem of latency is made significantly worse by the proliferation of Internet of Things devices and the accessing of real-time data. In order to solve this issue, the fog layer, which was previously an adjunct layer between the cloud layer and the user, is now being utilised. As the data could be retrieved from the fog layer even if it was close to the edge of the network, it made the experience more convenient for the user. The lack of security in the fog layer is going to be an issue. The simple access to sources provided by the fog layer architecture makes it vulnerable to a great number of assaults. Consequently, the purpose of this work is to build a seagull optimization-based feature selection approach with optimum extreme learning machine (SGOFS-OELM) for the purpose of intrusion detection in a fog-enabled WSN. The identification of intrusions in the fog-enabled WSN is the primary focus of the SGOFS-OELM approach that has been presented here. The given SGOFS-OELM strategy is designed to accomplish this goal by designing the SGOFS approach to choose the best possible subset of attributes. In this work, the ELM classification model is applied for the purpose of intrusion detection. In conclusion, the political optimizer (PO) is utilised in order to accomplish automatic parameter adjustment of the ELM technique, which ultimately leads to enhanced classification performance. In order to demonstrate the usefulness of the SGOFS-OELM approach, a number of simulations were carried out. As compared to the other benchmark models that were employed for this research, the suggested SGOFS-OELM models give the best accuracy, which is 99.97 percent. The simulation research demonstrates that the SGOFS-OELM approach has the potential to deliver a good performance in the intrusion detection process.

Keywords: fog computing; intrusion detection; machine learning; metaheuristics; security issues; wireless sensor networks

1 INTRODUCTION

The Internet of Things (IoT) is made up of many intelligent gadgets that are connected to one another all over the place. It is a network of the future generation that is always accessible and can communicate with one another without the need for human intervention [1, 2]. The devices generate a massive amount of data, which, in most cases, must first go through the computer processing stage before it can be used by analytical applications.

The processing of this data will naturally gravitate towards a cloud-related structure as the optimal option. A few examples of major applications that consume and analyse information from the Internet of Things include health care, intelligent pesticide control, smart grids, field surveillance, and transportation management [3]. In order to accommodate latency-sensitive analytical applications, fog-related computational structures proved to be effective. These structures helped deliver computational services as near as feasible to edge devices. Intrusion detection (ID) is a proactive security defensive system that monitors the running condition of a network and discovers intrusions, such as misoperations, as well as internal or external attacks [4], in order to allow the network to respond and intercept as required. The technology behind wired network ID has reached a mature stage and been divided into two categories: those categories are misuse-related and anomaly-related [5].

The acquisition of knowledge about assault technique and the prior definition of the intrusion mode were pre-requisites for the process of misappropriate usage recognition. The method of intrusion detection in fog-assisted wireless sensor networks is depicted in Fig. 1.

Fig. 1 shows an example of assisted fog-based intrusion detection. WSN

It is possible to determine whether an intrusion has occurred by determining whether or not the data characteristics that have been obtained match the database of intrusion patterns [6]. As a result, all it does is have a greater identification rate for specific assault methods, but it cannot be used for recognising attacks from unknown sources. A technology known as anomaly detection is being considered as a possible response to the continuously emerging new types of attacks [7]. This approach makes the presumption that cyber-attacks were relatively rare in comparison to other types of conduct. It is possible to identify that an intrusion has taken place by performing a comparison between typical paradigms and the behaviour of the network that has been recorded [8]. Anomaly detection deals with assaults that were not predicted; nonetheless, it has to train on more historical data in order to become effective.

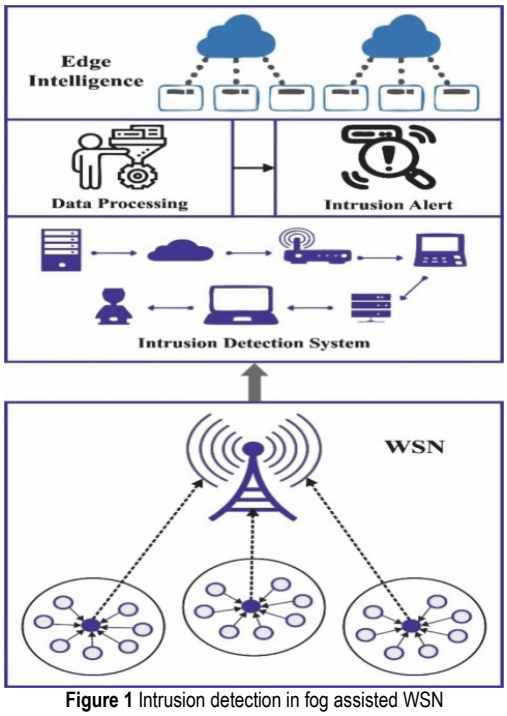

Figure 1 Intrusion detection in fog assisted WSN

It was predicted that the use of AI will improve the performance of the identification process. Numerous writers have made attempts to apply machine learning, artificial neural networks (ANNs), and evolutionary computing [9]. (ML) to the field of ID, and have arrived at fruitful conclusions as a result of their research. It is unable to use standard intrusion detection system (IDS) infrastructure due to the fact that WSN has its own benefits and downsides in terms of communication bandwidth, network scale, energy supply, processing capacity, networking mode, and storage [10]. Artificial intelligence technology often requires a large amount of processing power, in addition to requiring more electricity, more time to run, and more storage resources [11]. So, it is vital to make adjustments and alterations to the WSN-ID approach as per the original application instances and user demands and to seek for a balance among a variety of objectives, including energy consumption, security, and realtime.

In this study, a seagull optimization-based feature selection with optimal extreme learning machine (SGOFS-OELM) technique for intrusion detection in fog-enabled wireless sensor networks is developed. The identification of intrusions in the fog-enabled WSN is the primary focus of the SGOFS-OELM approach that has been presented here. In order to do this, the SGOFS-OELM strategy that is described proposes an approach based on SGOFS to choose the best possible subset of attributes. In this work, the ELM classification model is applied for the purpose of intrusion detection. In conclusion, the political optimizer (PO) is utilised in order to accomplish automatic parameter adjustment of the ELM technique, which ultimately leads to enhanced classification performance. A number of simulations were carried out in order to demonstrate that the SGOFS-OELM approach yielded superior results overall.

2 RELATED WORKS

Alzubi et al. [12] modelled an Effective Seeker Optimized method in conjunction with a ML-based IDS (ESOML-IDS) method for EC and FC atmospheres. The ESOML-IDS method mainly devises an innovative ESO-related FS technique for choosing the best subset of feature for detecting the existence of intrusion in EC and FC platforms. The authors even implemented a comprehensive learning PSO (CLPSO) with Denoising AE (DAE) for detecting intrusions. In [13], the authors examine ID approaches for mitigating assaults that use IoT security vulnerability. The authors introduced an ML-related two-layer hierarchical IDS that efficiently identifies intrusions in IoT networks whereas fulfilling the IoT resource limitations.

Reddy et al. [14] modelled a security process and guarantee truthful function of IoT networking with ID mechnaism. A network IDS was modelled on the basis of the conception of Exact Greedy Boosting ensemble technique for device execution in fog nodes since defending critical structure from accurate and timely recognition of malevolent actions. Sudqi Khater et al. [15] introduced a lightweight IDS related to a vector space representation utilizing a MLP approach. The authors assessed the IDS against ADFA-WD and ADFA-LD, which were novel generation mechanism calls data that uses and assaults several applications.

Abdel-Basset et al. [16] devised a forensics-related DL method (named Deep-IFS) for finding intrusions in IIoT traffic. A residual link among layers was devised to thwart information loss. One more challenge confronting the recent IIoT forensics structures is restrictive performance in managing Big IIoT traffic datasets made by IIoT gadgets and their limited scalability. This challenge can be solved by training and deploying the presented Deep-IFS in a fog computing platform. The authors [17] introduced an innovative lightweight IDS named sample selected ELM (SS-ELM). The reason why the authors devised "sample selected ELM" is that MEC hosts or fog nodes cannot save extremely large volumes of trained datasets.

Limitations of Existing Systems

- It can be seen from the above discussions that most of the security systems adopted across the fog assisted VANET environment are not distributed in nature.
- The implicit nature of VANET and in turn the data flow is distributed in nature. This makes it highly complex for deploying the existing security policies.
- Also, these security processes are less transparent.
- The data is highly mutable across the existing security policies which mean that any participating node can alter the data easily.
- At the same time, the existing security practices are highly resource intensive and incur high cost.
- The existing security policies are also tightly coupled with the participating nodes and this in turn defeats the fundamental distributed and loosely coupled nature of VANET.

3 THE PROPOSED MODEL

In this study, we have developed an automated intrusion detection approach, named SGOFS-OELM model in the fog enabled WSN. In the presented SGOFS-OELM technique, the major intention lies in the recognition of intrusions in the fog enabled WSN. To achieve this, the presented SGOFS-OELM technique carries out three processes such as SGO based feature subset selection, ELM classification, and PO based parameter tuning.

3.1 Algorithmic Process involved in SGOFS Technique

Primarily, the presented SGOFS-OELM technique designs SGOFS methodology to elect an optimal subset of features. SGO depends on the migrating and attacking performance of seagulls [18]. The arithmetical approach of attacking and migrating the prey was determined in the subsequent. The migrating process inspires the set of seagulls which is transfered in every place. At this point, the seagulls must fulfill 3 conditions:

For avoiding collision betwixt neighbors, a further parameter A was involved to assess a novel searching place:

$$\vec{C}_s = A \times \vec{P}_s(x) \tag{1}$$

in which x signifies the present iteration in the succeeding, $\vec{C}_s$ refers to the place of SA which could not collide with one another SA, $\vec{P}_s$ denotes the existing place of SA,

$$A = f_c - \left(x \times \left(f_c / Max_{\text{iteration}}\right)\right) \tag{2}$$

where: x = 0, 1, 2, ..., $Max_{\text{iteration}}$; whereas f_c control the frequency of A using slowly minimizing in f_c to 0. At this point, the value of f_c is set to 2. Next, to avoid the collision betwixt adjacent seagulls, the SA transfers to optimal neighbor way.

$$\vec{M}_s = B \times \left(\vec{P}_{bs}(x) - \vec{P}_s(x)\right) \tag{3}$$

Assume $\vec{M}_s$ be the location of SA $\vec{P}_s$ to optimal fit SA $\vec{P}_{bs}$ (viz., appropriate seagulls). The performance of B has been selected randomly using accountability to appropriate balance betwixt exploration as well as exploitation. B was evaluated as:

$$B = 2 \times A^2 \times rd \tag{4}$$

During the formula, rd symbolizes an arbitrary integer lies from zero and one. Finally, the SA upgrade their position near optimal SA.

$$\vec{D}_s = \left|\vec{C}_s + \vec{M}_s\right| \tag{5}$$

Consider $\vec{D}_s$ implies the distance betwixt the optimal appropriate SA and SA. The exploitation step concentrates on utilizing the experience and past searching procedure. The seagulls are the size for changing the angle and attack speed from the migrating. They recollect altitude with wing and weight. x, y, and z planes are provided in the subsequent

$$x' = r \times \cos(k) \tag{6}$$

$$y' = r \times \sin(k) \tag{7}$$

$$z' = r \times k \tag{8}$$

$$r = u \times e^{kv} \tag{9}$$

In which r depicts the radius of all the turns of spiral, k illustrates a random value in $[0 \le k \le 2A]$. u and v stand for constant to decide spiral shape, and e exemplifies base of natural logarithms. It can be evaluated in the subsequent formula:

$$\vec{P}_s(x) = \left(\vec{D}_s \times x' \times y' \times z'\right) + \vec{P}_{bs}(x) \tag{10}$$

At this point, $\vec{P}_s(x)$ staves optimum solution and upgrades the location of another SA.

The *fitness function* (FF) considered selected features and the classifier accuracy. It maximized classifier accuracy and shortened set size of selective attributes. Hence, the following *fitness function* (FF) was employed for assessing individual solution, as given in Eq. (11).

$$Fitness = \alpha * ErrorRate + (1 - \alpha) * \tag{11}$$

Here *Error Rate* means the classifier error rates utilizing selective attributes. *Error Rate* can be figured as percentage of incorrect classifiers (by ELM) to number of classifications done, stated as value between 0 and 1. $\#SF$ is the count of selected features and $\#All\_F$ represents total attributes in the actual data. α is employed for controlling the significance of subset length and classification quality. In our experiments, α is set to 0.9.

3.2 Intrusion Detection using Optimal ELM Model

For intrusion detection, ELM classification model is used in this study. According to feed-forward NN (FFNN), network infrastructure of ELM comprises output, input, and hidden layers (HL). An input weight and offsets are set arbitrarily, afterward the equivalent resultant weight can be attained [19]. Fig. 2 showcases the framework of ELM. At this point, considering that there are M instances (X_j, t_j), this NN representation of L HL nodes is as follows:

$$\sum_{I=1}^{l} \beta_i g\left(W_i \cdot OX_j + b_i\right) = y_j, j = 1 \ldots, M \tag{12}$$

whereas $g(x)$ denotes the activation function of HL, W_i signifies the input weighted but β_i is resultant weight, and b_i is offset of i-th HL. To attain the minimal resultant error is the purpose of single HL neural network (SLNN), viz.

$$\sum_{I=1}^{L} \beta_i g\left(W_i \cdot OX_j + b_i\right) = t_j, j = 1, \ldots, M \tag{13}$$

O denotes the outcome of HL node but H refers the chosen outcome:

$$O\beta = H \tag{14}$$

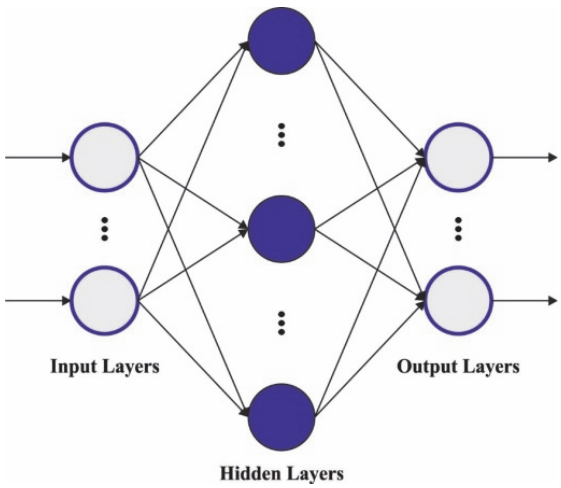

Figure 2 Architecture of ELM

Once the ELM was executed for training the SLNN, parameters W_i and β_i are arbitrary. If these 2 parameters were defined, resultant matrix T was solely attained, and trained of entire network is changed for solving the linear method for obtaining resultant weighted β, as follows:

$$\hat{\beta} = O^T H \tag{15}$$

In the last stage, PO technique is enforced for automated parameter tuning of ELM model, which results in improved classification performance. PO algorithm is inspired by the western political optimization technique that consists of 2 features [20]. The first assumption is, every citizen must improve their goodwill for winning the election. The next assumption is, every party must obtain additional seats in the parliament. PO has encompassed 5 phases such as election campaign, constituency allocation, interparty elections, parliamentary affairs, party formation, and party switching. The total population can be separated to n political parties.

$$P = \{P_1, P_2, P_3, \ldots, P_n\} \tag{16}$$

Every party consists of n party members shown in the following expression

$$P_i = \left\{p_i^1, p_i^2, p_i^3, \ldots, p_i^n\right\} \tag{17}$$

Every party member induces d dimension which can be shown in Eq. (18).

$$p_i^j = [p_{i,1}^j, p_{i,2}^j, p_{i,3}^j, \ldots, p_{i,d}^j]^T \tag{18}$$

All the solutions might be an election candidate. Consider n electoral district as demonstrated below.

$$C = \{C_1, C_2, C_3, \ldots, C_n\} \tag{19}$$

Assume n members in all the constituencies, as follows.

$$c_j = \left\{p_1^j, p_2^j, p_3^j, \ldots, p_n^j\right\} \tag{20}$$

The party leader can be defined by members with optimum fitness in party formulated by.

$$\begin{aligned} &q = \arg\min_{1\le j\le n} f\left(p_1^j\right), \forall i \varepsilon \{1, \ldots, n\} \\ &p_i^* = p_i^q \end{aligned} \tag{21}$$

Every party leader is demonstrated as follows.

$$P^* = \left\{p_1^*, p_2^*, p_3^*, \ldots, p_n^*\right\} \tag{22}$$

The winners of different constituencies are called Members of the Parliament, formulated by:

$$C = \left\{c_1^*, c_2^*, c_3^*, \ldots, c_n^*\right\} \tag{23}$$

Eqs. (24) and (25) are used to update the position of potential solutions in the election campaign stage.

$$p_{(i,k)}^j(z+1) = \begin{cases} \text{if } p_{(i,k)}^j(z-1) \le p_{(i,k)}^j(z) \le m^* \text{ or } p_{(i,k)}^j(z-1) \ge p_{(i,k)}^j(z) \ge m^*, \\ \qquad m^* + r(m^* - p_{(i,k)}^j(z)); \\ \text{if } p_{(i,k)}^j(z-1) \le m^* \le p_i k^j(z) \text{ or } p_{(i,k)}^j(z-1) \ge m^* \ge p_{(i,k)}^j(z) \\ \qquad m^* + (2r-1)\,|\, m^* - p_{(i,k)}^j(t)\,|; \\ \text{if } m^* \le p_{(i,k)}^j(z-1) \le p_{(i,k)}^j(z) \text{ or } m^* \ge p_{(i,k)}^j(z-1) \ge p_{(i,k)}^j(z) \\ \qquad m^* + (2r-1)\,|\, m^* - p_i k^l(z-1)\,|; \end{cases} \tag{24}$$

$$p_{(i,k)}^j(z+1) = \begin{cases} \text{if } p_{(i,k)}^j(z-1) \le p_{(i,k)}^j(z) \le m^* \text{ or } p_{(i,k)}^j(z-1) \ge p_{(i,k)}^j(z) \ge m^*, \\ \qquad m^* + r(m^* - p_{(i,k)}^j(z)); \\ \text{if } p_{(i,k)}^j(z-1) \le m^* \le p_i k^j(z) \text{ or } p_{(i,k)}^j(z-1) \ge m^* \ge p_{(i,k)}^j(z), \\ \qquad m^* + (2r-1)\,|\, m^* - p_{(i,k)}^j(t)\,|; \\ \text{if } m^* \le p_{(i,k)}^j(z-1) \le p_{(i,k)}^j(z) \text{ or } m^* \ge p_{(i,k)}^j(z-1) \ge p_{(i,k)}^j(z), \\ \qquad m^* + (2r-1)\,|\, m^* - p_i k^l(z-1)\,|; \end{cases} \tag{25}$$

Party switching is adapted for balancing exploitation and exploration. In the iteration process, an adaptive parameter λ is used, viz., considerably reduced from 1 to 0. Likewise, based on the likelihood λ, each candidate is selected and replaced with the worst member of arbitrarily elected party, as follows.

$$q = \arg\max_{i\le j\le n} f\left(p_i^j\right) \tag{26}$$

During election phase, the winners in the constituency are accomplished by using the following equation.

$$\begin{aligned} &q = \arg\min_{i\le j\le n} f\left(p_i^j\right) \\ &c_j^* = p_q^j \end{aligned} \tag{27}$$

The PO technique defines a FF to execute higher classifier results. It expresses a positive integer for representing the optimal act of candidate outcomes and is written in Eq. (28).

$$\begin{aligned} &fitness\left(x_i\right) = ClassifierErrorRate\left(x_i\right) = \\ &= \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \end{aligned} \tag{28}$$

4 EXPERIMENTAL VALIDATION

In this section, the intrusion detection results of the SGOFS-OELM approach are tested using the KDDCup99 dataset [21]. The dataset comprises 125973 samples with 41 attributes and two classes as depicted in Tab. 1.

Table 1 Detail on dataset

Descriptions	Values
No. of instances	125973
No. of attributes	41
No. of classes	2
Normal/Anomaly	67343/58630

The FS outcomes of the SGO-FS technique with other FS optimization methods in terms of best cost (BC) are studied in Fig. 3. The figure shows that the BGOA-V and BGOA methods have reported poor BC of 0.006763 and 0.006530 correspondingly. Then, the BGOA-S method has obtained certainly reduced BC of 0.004176. Although the TLBO-FS and GA-FS techniques have reported reasonable BC of 0.001108 and 0.001150, the SGO-FS method has least BC of 0.000981.

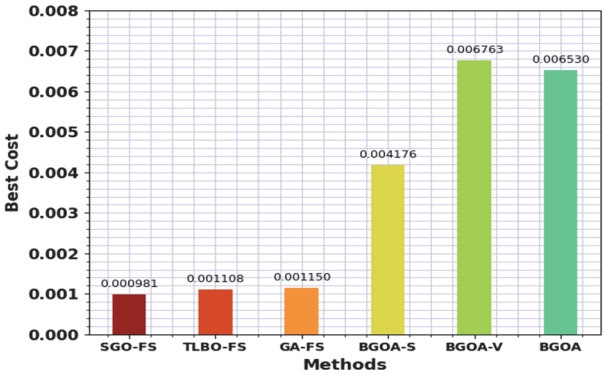

Figure 3 BC analysis of SGO-FS system with other approaches

In Tab. 2, the overall intrusion detection outcomes of the SGOFS-OELM method are examined under various attacks and different runs of execution. The results indicates that the SGOFS-OELM model has properly classified five distinct classes under all runs. On run-1, the SGOFS-OELM model has offered average $accu_y$ of 99.84%, $prec_n$ of 96.64%, $reca_l$ of 98.78%, F_{score} of 97.68%, and *MCC* of 97.57%. Meanwhile, on run-3, the SGOFS-OELM technique has rendered average $accu_y$ of 99.83%, $prec_n$ of 96.02%, $reca_l$ of 99.19%, F_{score} of 97.51%, and *MCC* of 97.42%. Eventually, on run-4, the SGOFS-OELM approach presented average $accu_y$ of 99.94%, $prec_n$ of 98.25%, $reca_l$ of 98.02%, F_{score} of 98.13%, and *MCC* of 98.08%. Furthermore, on run-5, the SGOFS-OELM method has offered average $accu_y$ of 99.97%, $prec_n$ of 99.70%, $reca_l$ of 97.78%, F_{score} of 98.69%, and *MCC* of 98.67%.

Table 2 Intrusion detection outcome of SGOFS-OELM approach with varying measures and runs

Class	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	*MCC*
Run-1					
Dos	99.76	99.67	99.66	99.67	99.48
R2l	99.92	93.05	97.89	95.41	95.4
Probe	99.81	99.06	98.88	98.97	98.87
U2r	100	91.67	97.78	94.62	94.67
Normal	99.71	99.75	99.71	99.73	99.42
Average	99.84	96.64	98.78	97.68	97.57
Run-2					
Dos	99.76	99.65	99.68	99.66	99.47
R2l	99.88	93.04	97.91	95.47	95.42
Probe	99.79	99.03	98.94	98.94	98.86
U2r	99.83	91.67	97.78	94.59	94.67
Normal	99.75	99.74	99.72	99.79	99.39
Average	99.8	96.63	98.81	97.69	97.56
Run-3					
Dos	99.71	99.58	99.64	99.61	99.38
R2l	99.94	94.36	97.87	96.08	96.07
Probe	99.78	98.91	98.74	98.82	98.7
U2r	100	87.5	100	93.33	93.54
Normal	99.71	99.76	99.69	99.72	99.41
Average	99.83	96.02	99.19	97.51	97.42
Run-4					
Dos	99.88	99.77	99.92	99.84	99.75
R2l	99.97	98.84	97.71	98.28	98.26
Probe	99.96	99.85	99.76	99.8	99.78
U2r	99.99	92.86	92.86	92.86	92.85
Normal	99.87	99.91	99.84	99.88	99.73
Average	99.94	98.25	98.02	98.13	98.08
Run-5					
Dos	99.94	99.8	99.88	99.84	99.75
R2l	99.99	98.99	99.32	99.15	99.15
Probe	99.96	99.8	99.8	99.8	99.78
U2r	100	100	90	94.74	94.87
Normal	99.95	99.93	99.87	99.9	99.78
Average	99.97	99.7	97.78	98.69	98.67

The TACC and VACC of the SGOFS-OELM methodology are investigated on intrusion detection performance in Fig. 4. The figure exhibites the SGOFS-OELM technique has shown improved performance with increased values of TACC and VACC. Notably, the SGOFS-OELM technique has reached maximum TACC outcomes.

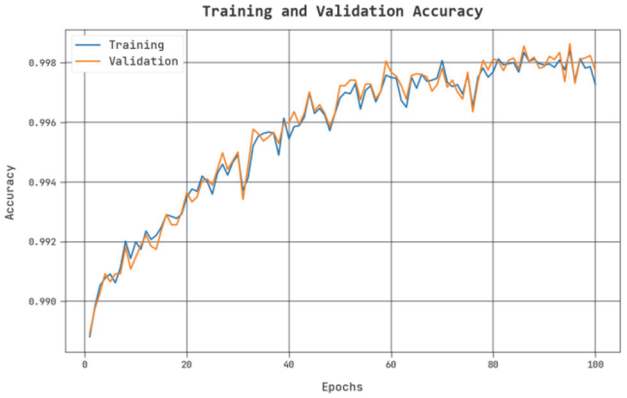

Figure 4 TACC and VACC analysis of SGOFS-OELM approach

Figure 5 TLS and VLS analysis of SGOFS-OELM system

The TLS and VLS of the SGOFS-OELM approach are tested on intrusion detection performance in Fig. 5. The

results show the SGOFS-OELM algorithm has revealed better performance with minimal values of TLS and VLS. In particular, the SGOFS-OELM approach has reduced VLS outcomes.

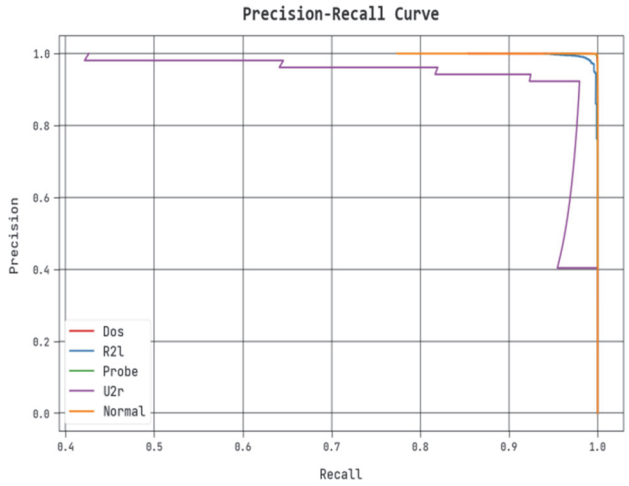

Figure 6 Precision-recall analysis of SGOFS-OELM approach

A clear precision-recall review of the SGOFS-OELM approach on test database is shown in Fig. 6. The results signify that the SGOFS-OELM method has improved values of precision-recall values.

The complete ROC study of the SGOFS-OELM method on test database is seen in Fig. 7. The figure exhibits the SGOFS-OELM approach and has shown its ability in classifying different classes under test database.

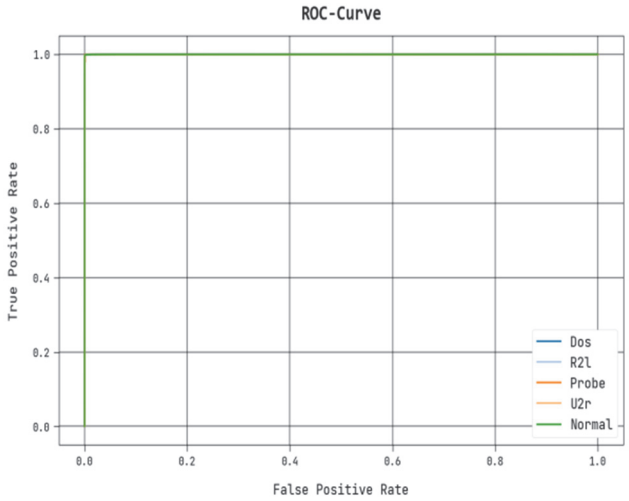

Figure 7 ROC analysis of SGOFS-OELM approach

In Tab. 3 and Fig. 8, an overall comparitive study of the SGOFS-OELM method with existing methods in terms of $accu_y$ is given [22-27]. The results indicate that the CS-PSO algorithm has reached reduced $accu_y$ of 75.51%. Next, the existing behaviour-IDS, COA-IDS, and DNN+SVM models have resulted in slightly enhanced $accu_y$ of 98.89%, 96.88%, and 92.03% respectively.

Although the DBN, MLIDS, and PSO-SVM models have managed to reach considerable outcomes with $accu_y$ of 99.96%, 99.93%, and 99.1% correspondingly, the SGOFS-OELM method has gained maximum performance with $accu_y$ of 99.97%. These results show the betterment of the SGOFS-OELM model on intrusion classification process.

Table 3 Comparison analysis of SGOFS-OELM approach with other algorithms

Methods	Accuracy
SGOFS-OELM	99.97
DBN Technique	99.96
MLIDS Algorithm	99.93
CS-PSO (2019)	75.51
PSO-SVM (2019)	99.1
Behavior-IDS (2019)	98.89
COA-IDS (2018)	96.88
DNN+SVM (2018)	92.03

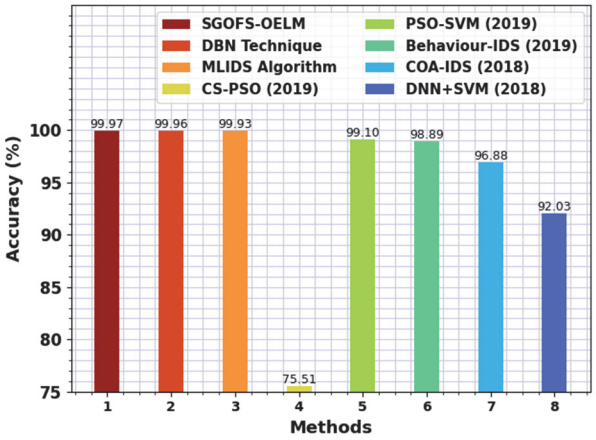

Figure 8 Accuracy analysis of SGOFS-OELM approach with other systems

5 CONCLUSION

In this research work, we came up with an automatic intrusion detection method that we gave the term SGOFS-OELM model in the fog enabled WSN. The identification of intrusions in the fog-enabled WSN is the primary focus of the SGOFS-OELM approach that has been presented here. In order to do this, the SGOFS-OELM strategy that is described proposes an approach based on SGOFS to choose the best possible subset of attributes. In this work, the ELM classification model is applied for the purpose of intrusion detection. At long last, the PO method is implemented in order to provide automatic parameter adjustment for the ELM approach, which ultimately leads to enhanced classification performance. A number of simulations had to be carried out before the SGOFS-OELM approach could demonstrate its superior performance. The simulation research demonstrates that the SGOFS-OELM approach has the potential to deliver a good performance in the intrusion detection process. In the future, the model that was provided may be expanded to incorporate the creation of strategies for energy management and predictive traffic flow in ITS. Also, lightweight authentication schemes with hybrid metaheuristics can be introduced to increase VANET's level of security.

6 REFERENCES

[1] Pundir, S., Wazid, M., Singh, D. P., Das, A. K., Rodrigues, J. J., & Park, Y. (2019). Intrusion detection protocols in wireless sensor networks integrated to Internet of Things deployment: Survey and future challenges. *IEEE Access*, *8*, 3343-3363. https://doi.org/10.1109/ACCESS.2019.2962829

[2] Wang, N., Li, X. J., & Nie, H. (2021). Digital Production Control of Manufacturing Workshop Based on Internet of Things. *International Journal of Simulation Modelling*, *20*(3), 606-617. https://doi.org/10.2507/IJSIMM20-3-CO15

[3] Abdussami, A. A. & Farooqui, M. F. (2021). Incremental deep neural network intrusion detection in fog based IoT environment: An optimization assisted framework. *Indian Journal of Computer Science and Engineering*, *12*(6), 1847-1859. https://doi.org/10.21817/indjcse/2021/v12i6/211206191

[4] Fang, W., Zhang, W., Chen, W., Liu, Y., & Tang, C. (2020). TMSRS: trust management-based secure routing scheme in industrial wireless sensor network with fog computing. *Wireless networks*, *26*, 3169-3182. https://doi.org/10.1007/s11276-019-02129-w

[5] Sahar, N., Mishra, R., & Kalam, S. (2021). Deep learning approach-based network intrusion detection system for fog-assisted IoT. *Proceedings of international conference on big data, machine learning and their applications*, 39-50.

[6] Rahman, M. A., Asyhari, A. T., Leong, L. S., Satrya, G. B., Tao, M. H., &Zolkipli, M. F. (2020). Scalable machine learning-based intrusion detection system for IoT-enabled smart cities. *Sustainable Cities and Society*, *61*, 102324. https://doi.org/10.1016/j.scs.2020.102324

[7] Saif, S., Das, P., Biswas, S., Khari, M., & Shanmuganathan, V. (2022). HIIDS: Hybrid intelligent intrusion detection system empowered with machine learning and metaheuristic algorithms for application in IoT based healthcare. *Microprocessors and Microsystems*, 104622. https://doi.org/10.1016/j.micpro.2022.104622

[8] Vaiyapuri, T., Sbai, Z., Alaskar, H., & Alaseem, N. A. (2021). Deep learning approaches for intrusion detection in IIoT networks-opportunities and future directions. *International Journal of Advanced Computer Science and Applications*, *12*(4). https://doi.org/10.14569/IJACSA.2021.0120411

[9] Ramkumar, M. P., Daniya, T., Paul, P. M., & Rajakumar, S. (2022). Intrusion detection using optimized ensemble classification in fog computing paradigm. *Knowledge-Based Systems*, *252*, 109364. https://doi.org/10.1016/j.knosys.2022.109364

[10] de Souza, C. A., Westphall, C. B., Machado, R. B., Loffi, L., Westphall, C. M., & Geronimo, G. A. (2022). Intrusion detection and prevention in fog based IoT environments: A systematic literature review. *Computer Networks*, 109154. https://doi.org/10.1016/j.comnet.2022.109154

[11] Chang, V., Golightly, L., Modesti, P., Xu, Q. A., Doan, L. M. T., Hall, K., Boddu, S., & Kobusińska, A. (2022). A Survey on Intrusion Detection Systems for Fog and Cloud Computing. *Future Internet*, *14*(3), 89. https://doi.org/10.3390/fi14030089

[12] Alzubi, O. A., Alzubi, J. A., Alazab, M., Alrabea, A., Awajan, A., & Qiqieh, I. (2022). Optimized Machine Learning-Based Intrusion Detection System for Fog and Edge Computing Environment. *Electronics*, *11*(19), 3007. https://doi.org/10.3390/electronics11193007

[13] Roy, S., Li, J., & Bai, Y. (2022). A Two-layer Fog-Cloud Intrusion Detection Model for IoT Networks. *Internet of Things*, 100557. https://doi.org/10.1016/j.iot.2022.100557

[14] Reddy, D. K. K., Behera, H. S., Nayak, J., Naik, B., Ghosh, U., & Sharma, P. K. (2021). Exact greedy algorithm based split finding approach for intrusion detection in fog-enabled IoT environment. *Journal of Information Security and Applications*, *60*. https://doi.org/10.1016/j.jisa.2021.102866

[15] Sudqi Khater, B., Abdul Wahab, A. W. B., Idris, M. Y. I. B., Abdulla Hussain, M., & Ahmed Ibrahim, A. (2019). A lightweight perceptron-based intrusion detection system for fog computing. *Applied sciences*, *9*(1), 178. https://doi.org/10.3390/app9010178

[16] Abdel-Basset, M., Chang, V., Hawash, H., Chakrabortty, R. K., & Ryan, M. (2020). Deep-IFS: intrusion detection approach for industrial internet of things traffic in fog environment. *IEEE Transactions on Industrial Informatics*, *17*(11), 7704-7715. https://doi.org/10.1109/TII.2020.3025755

[17] An, X., Zhou, X., Lü, X., Lin, F., & Yang, L. (2018). Sample selected extreme learning machine based intrusion detection in fog computing and MEC. *Wireless Communications and Mobile Computing*, *2018*. https://doi.org/10.1155/2018/7472095

[18] Dhiman, G., Singh, K. K., Slowik, A., Chang, V., Yildiz, A. R., Kaur, A., & Garg, M. (2021). EMoSOA: a new evolutionary multi-objective seagull optimization algorithm for global optimization. *International Journal of Machine Learning and Cybernetics*, *12*(2), 571-596. https://doi.org/10.1007/s13042-020-01189-1

[19] Qi, H., Xie, S., Chen, Y., Wang, C., Wang, T., Sun, B., & Sun, M. (2022). Prediction Methods of Common Cancers in China using PCA-ANN and DBN-ELM-BP. *IEEE Access*, *10*. https://doi.org/10.1109/ACCESS.2022.3215706

[20] Askari, Q. & Younas, I. (2021). Political optimizer based feed forward neural network for classification and function approximation. *Neural Processing Letters*, *53*(1), 429-458. https://doi.org/10.1007/s11063-020-10406-5

[21] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[22] Maheswari, M. & Karthika, R. A. (2021). A Novel QoS Based Secure Unequal Clustering Protocol with Intrusion Detection System in Wireless Sensor Networks. *Wireless Personal Communications: An International Journal*, *118*(2), 1535-1557. https://doi.org/10.1007/s11277-021-08101-2

[23] Kumar, G., Mohan, S., & Nagesh, A. (2021). An Ensemble of Feature Subset Selection with Deep Belief Network Based Secure Intrusion Detection in Big Data Environment. *Indian Journal of Computer Science and Engineering (IJCSE)*, *12*(2), 409-420. https://doi.org/10.21817/indjcse/2021/v12i2/211202101

[24] Li, J., Zhao, Z., Li, R. et al. (2018). AI-based two-stage intrusion detection for software defined IoT networks. *IEEE Internet Things Journal*, *6*(2), 2093-2102. https://doi.org/10.1109/JIOT.2018.2883344

[25] Diro, A. A. & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for internet of things. *Future Generation Computer Systems*, *82*, 761-768. https://doi.org/10.1016/j.future.2017.08.043

[26] Yang, Y., Zheng, K., Wu, C. et al. (2019). Building an effective intrusion detection system using the modified density peak clustering algorithm and deep belief networks. *Applied Sciences*, *9*(2), 238-244. https://doi.org/10.3390/app9020238

[27] Djenouri, Y, Belhadi, A., Lin, J. C. W. et al. (2019). Adapted k-nearest neighbors for detecting anomalies on spatio-temporal traffic flow. *IEEE Access*, *7*, 10015-10027. https://doi.org/10.1109/ACCESS.2019.2891933

Contact information:

Thiruppathi MUTHU, Assistant Professor
Department of Electronics and Communication Engineering,
Vivekanandha College of Engineering for Women,
Tiruchengode, Tamil Nadu, India
E-mail: mailtothiruppathi@gmail.com

Dr. Vinoth Kumar KALIMUTHU, Professor
(Corresponding Author)
Department of Electronics and Communication Engineering,
SSM Institute of Engineering and Technology, Dindigul, Tamil Nadu, India
E-mail: vinodkumaran87@gmail.com