International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I6P208
Volume 70 Issue 6, 63-72, June 2022

Original Article

Optimized Detector Generation Procedure for Wireless Sensor Networks based Intrusion Detection System

Giribabu Sadineni[1], M. Archana[2], Rama Chaithanya Tanguturi[3]

[1]*Department of Computer Science & Engineering, Annamalai University, Tamil Nadu, India.*
[2]*Department of Information Technology, Annamalai University, Tamil Nadu, India.*
[3] *Department of Computer Science & Engineering, PACE Institute of Technology and Sciences, Andhra Pradesh, India.*

[1]sadinenigiri1521@gmail.com

Received: 27 March 2022 Revised: 23 May 2022 Accepted: 31 May 2022 Published: 27 June 2022

Abstract *- Wireless Sensor Networks (WSNs) performance has been degraded by the intrusion and eliminating the capability to implement its functions. Through several aspects of WSNs, real-time protection is the main crucial component according to the increasing amount of cyber threats. Moreover, the latest devices have reduced security features and are vulnerable to cyber-attacks. It is very crucial to construct the system to detect real-time intrusion detection. In this paper, an optimized detector generation procedure (ODGP) is proposed and classified using the weighted SVM optimizer. The proposed technique is constructed to improve the accuracy of intrusion detection, improved detection rate, reduce false alarm rate, and also minimize the execution time than the existing techniques using the CICIDS2017 dataset. The performance evaluation results proved that the proposed ODGP technique enhances the performance that other related techniques.*

Keywords *- Wireless Sensor Networks (WSNs), Intrusion Detection System, Detector generation procedure, CICIDS2017 dataset, Weighted SVM.*

1. Introduction

The rapid improvements in wireless transmission, micro-electronics, and Wireless sensor networks (WSNs) through the improved characteristic development [1]. The WSNs have been utilized in several real-time applications like health care, industry, environment, and military [2]. The dynamically configured routing with many monitoring of the data flow and the solution to provide the safety-related issues of WSNs [3]. The enhanced encryption methodologies utilize the key management and the response functionality of intrusion detection that could be utilized in the dissimilar network layers and the enhanced techniques [4].

Conventional networks have the resources like system logs, files, and processors, which could not be utilized in WSNs and should be applied to demonstrate the feature data for performing the process of intrusion detection [5]. There are several attacks in WSNs, dissimilar to conventional networks, can improve the intrusion detection system for identifying unknown attacks [6]. The proposed algorithm must be constructed according to the network requirements whenever the WSNs have restricted resources like bandwidth, energy utilization, and storage space. The restricted storage space is very hard to store a huge amount of sensor nodes.

Normally, the opportunistic routing needs to detect the Denial of Service attacks in which invalid data must be forwarded to the recipient node through several forwarders, which must be completed with hypothetical analysis [7]. The requirement of the security verification technique to secure this kind of attack, the network system could be ensured the data packets are delivered from the sensor nodes and that they cannot be changed by the aggressors while communicating [8]. Additionally, a digital signature structure for completing the validation must enhance the computational ratio of the particular node in the network and expand the packet delivery [9]. The client verification technique for wireless sensor networks has been validated to provide efficiency [10]. The proposed technique has the framework to enhance the authentication procedure by enhancing the network performance to reduce the network traffic, improve network lifetime, and secure the nodes from several kinds of attacks by improving the execution of the system performances.

The proposed technique is constructed to disconnect denial of service aggregators in wireless sensor networks with opportunistic routing that can enhance the requirement of candidate forwarders along with the particular data packet verification. The re-establishment of the candidate nodes is used to maintain the integrity and reliability of data for recursive verification. The intrusion detection system is constructed according to the knowledge needed for storing many formatted intrusion patterns. The system identifies the intrusion through

pattern matching. The libraries store the behavior characteristics due to the huge amount of computation required for constructing the intrusion detection system, which also needs some more energy. The proposed algorithm must be considered the communication overhead, which needs to be reduced. The proposed technique is constructed to produce the minimum transmission overhead, reduce network resources, and improve the data communication rate.

2. Related works

The detection technique initially monitors the behaviors in the specific technique due to the system's specifications [11]. The abnormal condition is observed as the deviation from the common behavior that the behaviors are demonstrated as manually, so any modification of the specific behavior will enhance the computing time [12]. The masquerade attacks are identified using mutual protection as it checks the total packets delivered successfully by the specific node. This technique cannot be implemented whenever there is a low transmission range [13]. The distributed technique has been constructed with predefined rules where the monitoring nodes are identified in the system, that the characteristics of the defined attacks through the distribution process and the resource utilization of monitoring nodes are very high [14].

The cluster-based intrusion detection system has been implemented for prevention and early detection. The hybrid hierarchy architecture constructs the intrusion detection system according to the deployment of the sensors with cluster heads. The exceptions have been identified with the specification-related technique through the central server [15]. The misuse detection system identifies the known attacks according to the intrusion detection. The propagation properties of wireless transmission and the nodes are largely distributed with the expanded security system that can be utilized in WSNs. The newly designed framework contains the global and local agents that every node checks whether the data flow is normal or not for the adjacent nodes.

The spontaneous watchdog technique is used to identify the global agents to ensure a small number of agents within the network. This technique has the drawback of packet collisions, the randomness of the global proxy, and the abnormal detection demonstrates whenever the intrusion occurs through learning the common behavior of the sensor nodes [16]. The statistical model is implemented for detecting the resource depletion attacks that every node monitors the mean receiving rate of packets from adjacent nodes to construct the statistical model. The nearest packets of adjacent nodes are utilized for performing the statistical analysis that the fixed-width clustering technique has been constructed to identify the common behaviors. It generates a group of clusters in the feature space [17].

The exception is identified through the particular threshold that every sample is compared with the cluster to determine if it is abnormal. This model is deployed on every node that consumes a group of computational resources [18]. Resource consumption is the main issue whenever the network size is very large; gathering and training information on every sensor node is not feasible. The isolation table is generated for constructing the intrusion detection model in that the table stores the exception data, and the detection agent utilizes it for segregating the suspect nodes [19]. The isolation tables are generated from the cluster heads and are used for monitoring the sensor nodes. The learning automata technique has been constructed for intrusion detection that this technique is utilized the sample data packets and identifies the malicious related nodes. A low-complexity model combines energy-sensing through the random learning automata and sampling technique to perform the energy utilization.

3. Proposed work

While performing the data aggregation, the node generates the data packets and forwards them to the sink node, and this procedure is repeated whenever the network lifetime has been improved. Communication reliability is utilized to monitor the success rate while achieving the quality of service for forwarding the data packets. The packets are delivered from one node to the recipient node; the communication reliability is computed in Eq. (1).

$$\delta_i = 1 - \left(1 - \gamma_{ij}\right)^{\alpha} \tag{1}$$

The entire communication reliability is computed in Eq. (2).

$$\delta_i = \prod_k \left(1 - \left(1 - \gamma_{ij}\right)^{\alpha}\right) \tag{2}$$

Whenever the computed value is at the highest level, and communication reliability is also at the highest level, the data aggregation technique is utilized for gathering the data from the sensor nodes to the sink node through the highest amount of load that will form the data aggregation. The data aggregation is computed as whenever the data has been delivered to the sink node that data is utilized for aggregation procedure which the intermediate node aggregation than the remaining nodes. The intermediate node aggregation value is computed in Eq. (3).

$$\varphi(i,j) = \pi\left(\beta_i + \varphi_j\right) \tag{3}$$

Where π is the compression ratio, it is directly related to the data collection; the intermediate nodes are transmitting several packets regularly within the window size of α. The sensor nodes have not received the acknowledgment for delivering packets that the receiver generates several copies of the specific data stored. The repeated data has been eliminated, and the whole aggregation procedure has been finished. The packets can

hold a lot of data, which brings data overloading. The nodes can be transmitted through the sink nodes, and the reduced data copies are eliminated automatically.

The intermediate nodes can perform the data aggregation process while the data packets are removed despite unreliable links. The efficient communication can be performed through the residual energy that the nodes can save the energy while relaying. The distance is computed in Eq. (4).

$$Dist(i) = \frac{Ed_{i,s}}{Radius} \quad (4)$$

Where Ed is the Euclidean distance from node i to sink node s and radius is measured with the transmission range, the communication reliability is computed in Eq. (5).

$$Com_{reliablity} = \delta \quad (5)$$

The senor has the largest packet delivery details as several copies for performing the packet aggregation to eliminate data loss and increase communication reliability.

Algorithm 1 – Establishment of the network

Begin Procedure Network_Establishment ( )

Initialization of communication with reliability

For k = 0 to 1 do

every node having the initial energy

Assign receiver nodes

Identify the nodes having the highest energy level among adjacent hops

Calculate communication reliability and distance

Complete the communication process through adjacent hops regularly

The intermediate nodes saved the delivered packets

Remove the redundant copies

Complete data aggregation

End For

End Procedure

The Eigenvector is demonstrated as $\alpha = \{\alpha_1, \alpha_2, \dots\dots, \alpha_n\}$, where n is the dimension of the system that every vector feature is illustrated in the real-time interval [0,1], and the whole state space is demonstrated as $\delta = [0,1]^n$. It is segregated into identity and $non_{identity}$. In constructing an intrusion detection system, the identity is related to the normal system and $non_{identity}$ It is related to the abnormal system in Eq. (6) and Eq. (7).

$$identity \cap non_{identity} = \emptyset \quad (6)$$

$$identity \cup non_{identity} = Full \quad (7)$$

Fully demonstrates the whole samples of the problem set that the detector set contains similar structures as the Full values and the detector position and radius. The function $fn\,(Full, rad)$ illustrates the affinity within the full and radius degrees with data structures; the Euclidean distance within the Eigenvectors is computed in Eq. (8).

$$fn\,(Full, rad) = \sqrt{\sum_{i=1}^{n} (fun.x_i - rad.y_i)^2} \quad (8)$$

The generation of the particular detectors has the condition in Eq. (9).

$$fn\,(Full, rad) \leq rad_s + rad_d \quad (9)$$

The detection procedure has the set of detectors with the following condition in Eq. (10).

$$fn\,(Full, rad) \leq rad_d \quad (10)$$

The intrusion detection system has been constructed transparent and sustainable; it should be very easy to deploy. The WSN infrastructure has a hierarchical framework with clusters that consist of the nodes which transmit the data packets to the base station. This framework reduces the energy utilization on the whole network nodes, diminishes the transmission burden, and extends the network lifetime by utilizing the hierarchical framework.

Algorithm 2: Detector Generation procedure

Input: Training set

Output: the set of detectors

Step 1: Initialization of the training set

Step 2: Identify the detector $detect_{new}$ through random values

Step 3: Compute the Euclidean distance within the $detect_{new}$ into the training set

Step 4: If $fn\,(Full, rad) \leq rad_s + rad_d$ is true, then execute step 2; otherwise, process step 3

Step 5: Include $detect_{new}$ to the set of detectors

Step 6: If the detector size is set > threshold value, exit the process; otherwise, process step 2.

Step 7: Identify the candidate detector set through subspaces.

Step 8: Divide the space using the training set.

Step 9: Determine the candidate detector set is the final set of detectors; otherwise, execute step 7.

This procedure demonstrates the detector coverage to the sub spaces while performing the detector generation procedure. The WSNs have a restricted amount of nodes, and the computation cost is not useful for maintaining the data aggregation procedure, so the proposed technique is constructed to perform the enhanced data aggregation process. The proposed technique has enhanced intrusion detection by initially analyzing the distribution of the identity set in the network deployment and then segregating the network into several sub-groups. The randomly constructed detection technique must be generated through the detected intrusion and minimizes the time cost of distance computation. In the detection procedure, both the identity and $non_{identity}$ States are activated whenever the Full detector set has been examined in Fig. 1.

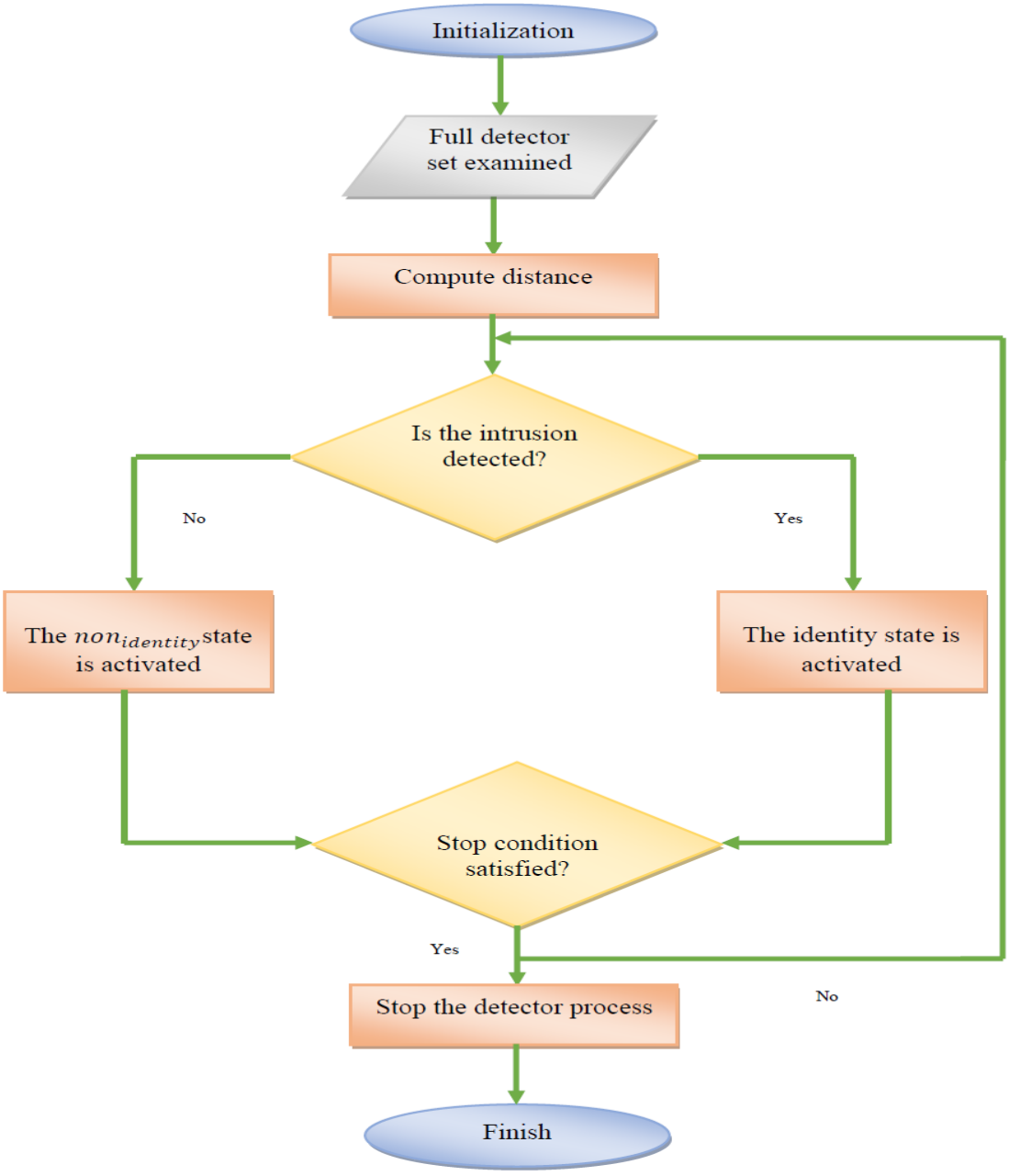

Fig. 1 The detection procedure

The dataset performs the decoding technique to estimate the dataset's contents and perform the normalization with Eq. (11).

$$\delta' = \frac{(Exact_{value} - Low_{value})}{(High_{value} - Low_{value})} \quad (11)$$

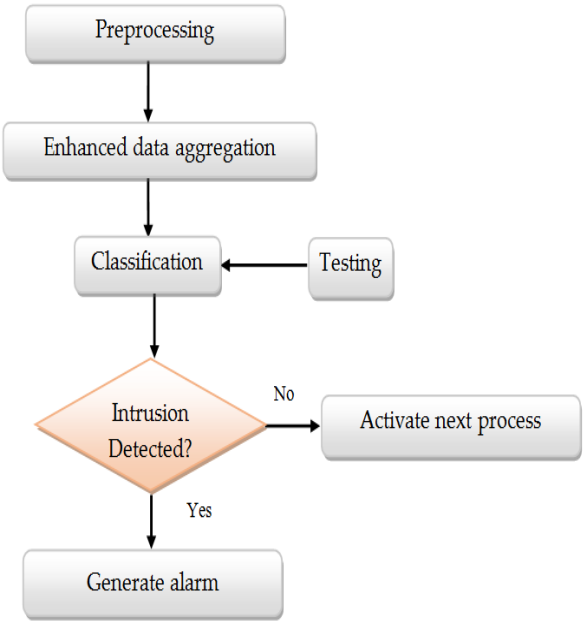

Fig. 2 Block diagram of the proposed technique

After completing the pre-processing through the dataset and segregating the feature vector, the training dataset is divided into clusters in that every cluster has the separation of normal and abnormal behaviors. The feature vector has been received after the training state; the enhanced classification technique has been produced the decision procedure from the clustering output with a reduced amount of outliers for an enhanced detection rate. The proposed technique has optimized the classification for producing the optimal decision output. Every vector in the dataset has been classified into the normal or abnormal state through the decision function, and the block diagram of the proposed technique is illustrated in Fig. 2.

The weighted SVM has the whole features for generating the decision through the weights that the optimization values are established in the training state and have been utilized for the decision model while classification. The correlation degree is computed through the vectors in Eq. (12)

$$\gamma(\alpha_0, \alpha_i) = \frac{1}{n}\sum_{j=1}^{n} \gamma(\alpha_0(j), \alpha_i(j)) \quad (12)$$

The value of weight is computed in Eq. (13).

$$W_i = \frac{1}{\sum_{k=1}^{n} \gamma_k |\, D_k(i) - D_k(0)| + \varepsilon} \quad (13)$$

Where ε is the smallest constant value, $D(i)$ is the normalization data, and the weighted kernel value (WK) is computed in Eq. (14).

$$WK(\alpha^i, \alpha^j) = \exp\left(-\gamma ||\alpha^i - \alpha^j||_2^2\right) \quad (14)$$

The optimized solution of the proposed technique indicates the minimum values of iteration, and the decision function is computed in Eq. (15).

$$DF = \frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\gamma_j WK(\alpha^i, \alpha^j)\right) \quad (15)$$

Fig. 3 demonstrates the weighted SVM flowchart in that the feature vector is constructed through the pre-processing data function, and the output class is produced through the components and weighted values and bias.

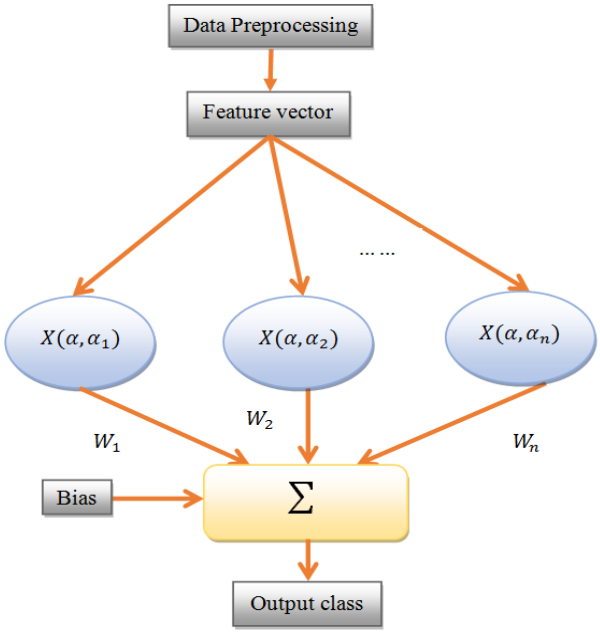

Fig. 3 Weighted SVM flowchart

4. Performance Evaluation

The proposed ODGP technique is compared with the related techniques of APAE [20], NDAE [21], and MSML [22] with the performance parameters of Accuracy, Precision, Recall, and F score, detection rate, false alarm rate, confusion matrix and execution time. The experiments have been conducted using Matlab R2019b, a commonly used modelling tool in several applications, and the experimental parameters are illustrated in Table 1.

Table 1. Parameter settings

Parameter	value
Simulation size	1000 m x 1000 m
Time	10200 s
The total amount of nodes	1000
Attackers type	11
Model for mobility	RWP
Pause period	8 s
Size of the cluster	15
Maximum speed	11 m/s
Simulation	Matlab R 2019b
Operating system	Windows 10 (64 bit)
CPU	4 cores
Speed	3.2 GHz

CICIDS2017 dataset [23] has been utilized to experiment with attack detection with seven classes; a total of 15,023 parameters have been used to produce the classification result. The class-wise parameters have been identified as the data distribution with several classes in the dataset dissimilarly.

A confusion matrix is a table that demonstrates the classification performance-based visualization for the proposed technique. It generates an easy analysis to illustrate the technique that has confused within the classes and unconfused with another one. In the table, every column and row demonstrate the class, and the index value (x, y) demonstrates the total amount of instance data that belongs to the x^{th} class, the predicted value belongs to the y^{th} class. It is easier to inspect the errors through the table as the exact predictions are on the diagonal of the table, and the outside values are demonstrated on the errors. Fig. 4 demonstrates the confusion matrix for the CICIDS2017 dataset.

Predicted Label

True Label / Class	Benign	Portscan	DoS	Web Attack	Infiltration	Brute Force	Bot
Benign	1875	680	233	12	405	125	28
Portscan	38	2310	560	0	8	0	0
DoS	135	205	3150	0	10	0	2
Web Attack	21	4	0	760	0	0	2
Infiltration	6	0	0	0	32	0	0
Brute Force	10	4	2	0	0	4214	0
Bot	93	0	0	0	0	0	565

Fig. 4 Confusion matrix

The precision value is computed in Eq. (16).

$$pre_{value} = \frac{True_{+ve}}{True_{+ve} + False_{+ve}} \qquad (16)$$

where $True_{+ve}$ is the total amount of values that are correctly classified, $False_{+ve}$ is the total amount of values which is wrongly classified. $False_{-ve}$ is the total amount of values from one class that are wrongly classified into another class.

The recall value is computed in Eq. (17).

$$recall_{value} = \frac{True_{+ve}}{True_{+ve} + False_{-ve}} \quad (17)$$

The F_{score} is computed in Eq. (18).

$$F_{score} = 2\ x\ \left(\frac{pre_{value}\ x\ recall_{value}}{pre_{value} + recall_{value}}\right) \quad (18)$$

The accuracy is the measurement of classified data, computed in Eq. (19).

$$Accuracy = \frac{True_{+ve} + True_{-ve}}{True_{+ve} + True_{-ve} + False_{+ve} + False_{-ve}} \quad (19)$$

Fig. 5 demonstrates the performance evaluation with various accuracy-related parameters of the proposed technique with the related techniques, and the result proves that the proposed technique has a better performance.

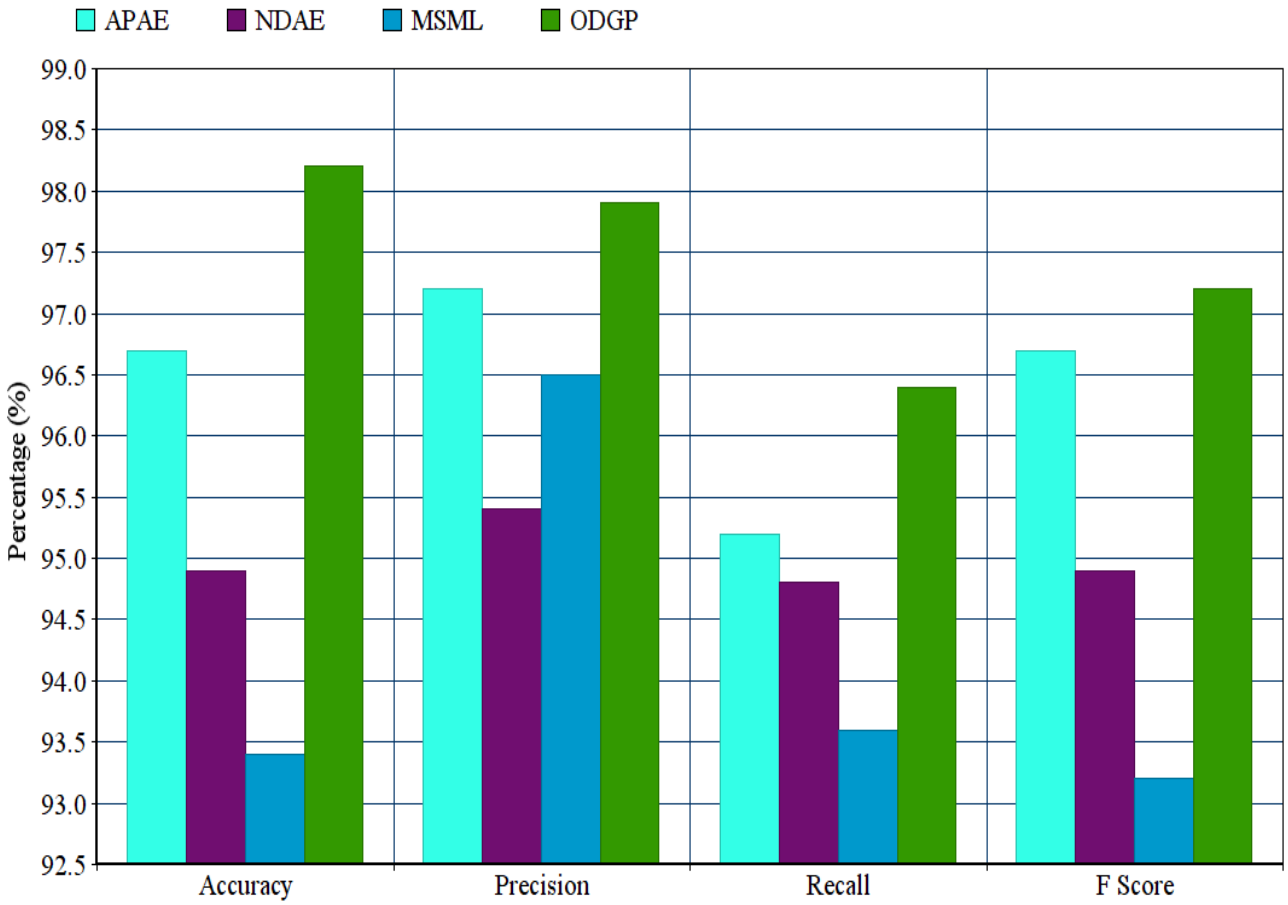

Fig. 5 Performance analysis

The total number of features is the average number of identified features for effectively classifying the behaviors. The execution time is the measurement of the time required to complete the classification process in Eq. (20).

$$Exec_{time} = Completed_{time} - beginning_{time} \quad (20)$$

Fig. 6 demonstrates the execution time for the proposed technique compared with the related techniques, and the result illustrates that the proposed technique has a minimized execution time compared with the other techniques.

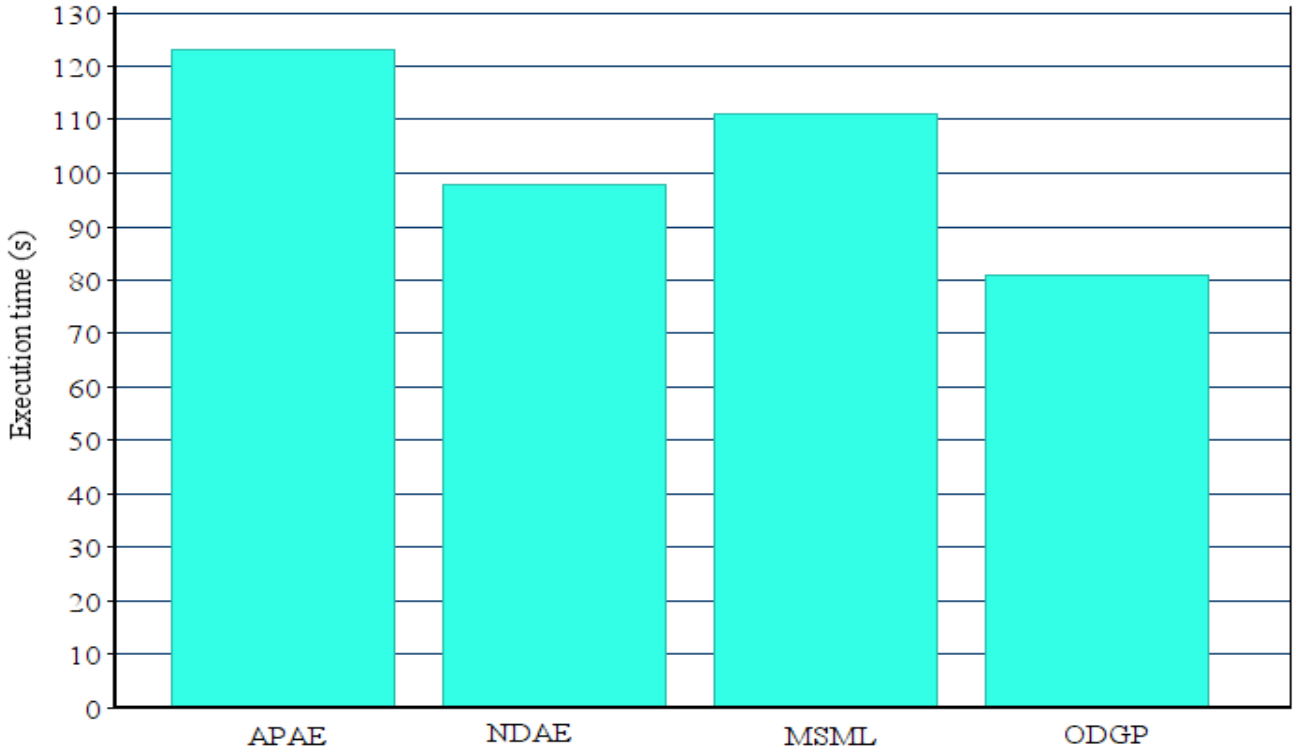

Fig. 6 Execution Time

The detection rate has been reduced whenever the number of attackers is increased rapidly. Several attackers and the sensor nodes have an abnormal state, so the detection rate has been minimized. While the time interval within attacks is higher and the behavior is lesser in the coverage range, the detection rate is demonstrated in Fig. 7.

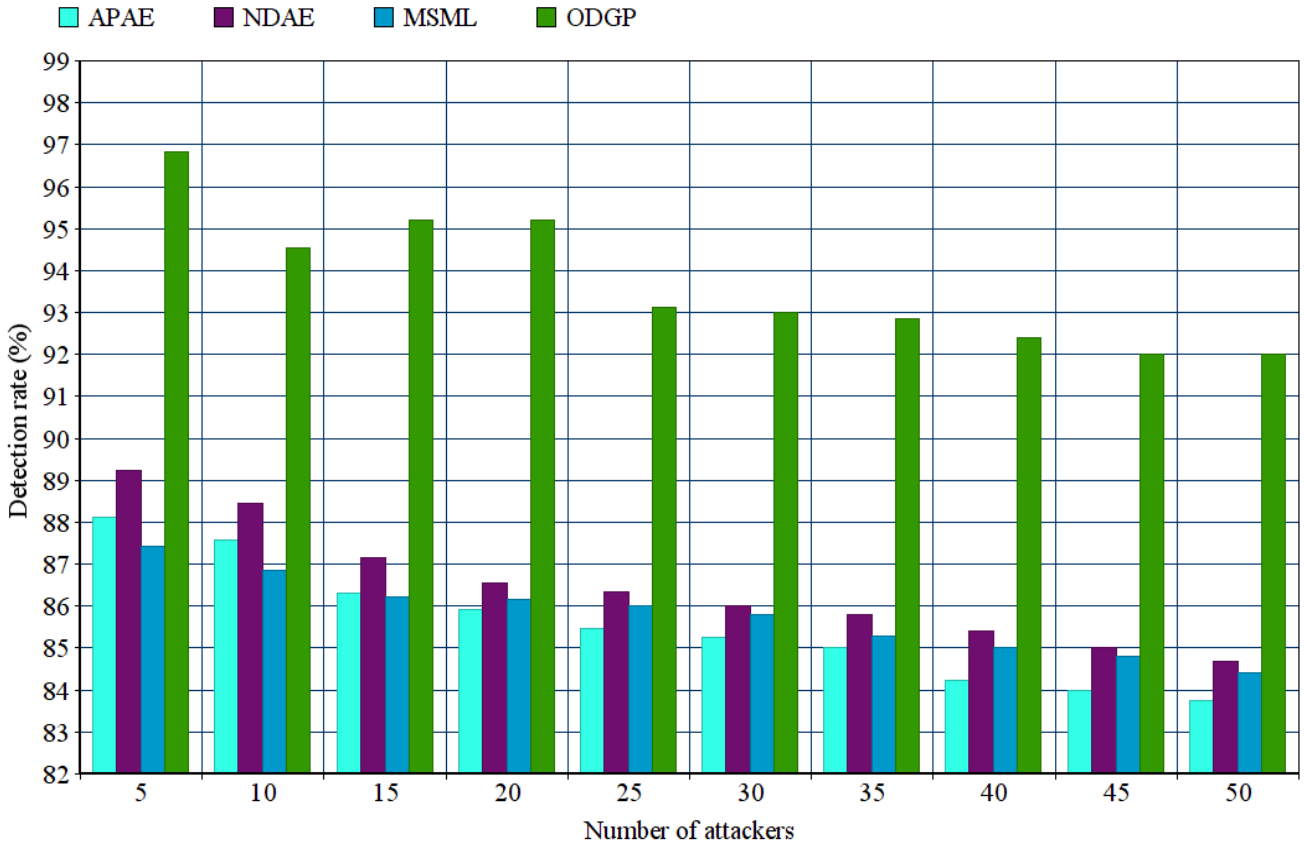

Fig. 7 Detection rate

The false alarm rate is computed in Eq. (21).

$$False_{alarm} = \frac{False_{+ve}}{True_{-ve} + False_{+ve}} \qquad (21)$$

The performance analysis for the False alarm rate evaluation parameter is demonstrated in Fig., and the result proves that the proposed technique has a minimized false alarm rate compared with the related techniques.

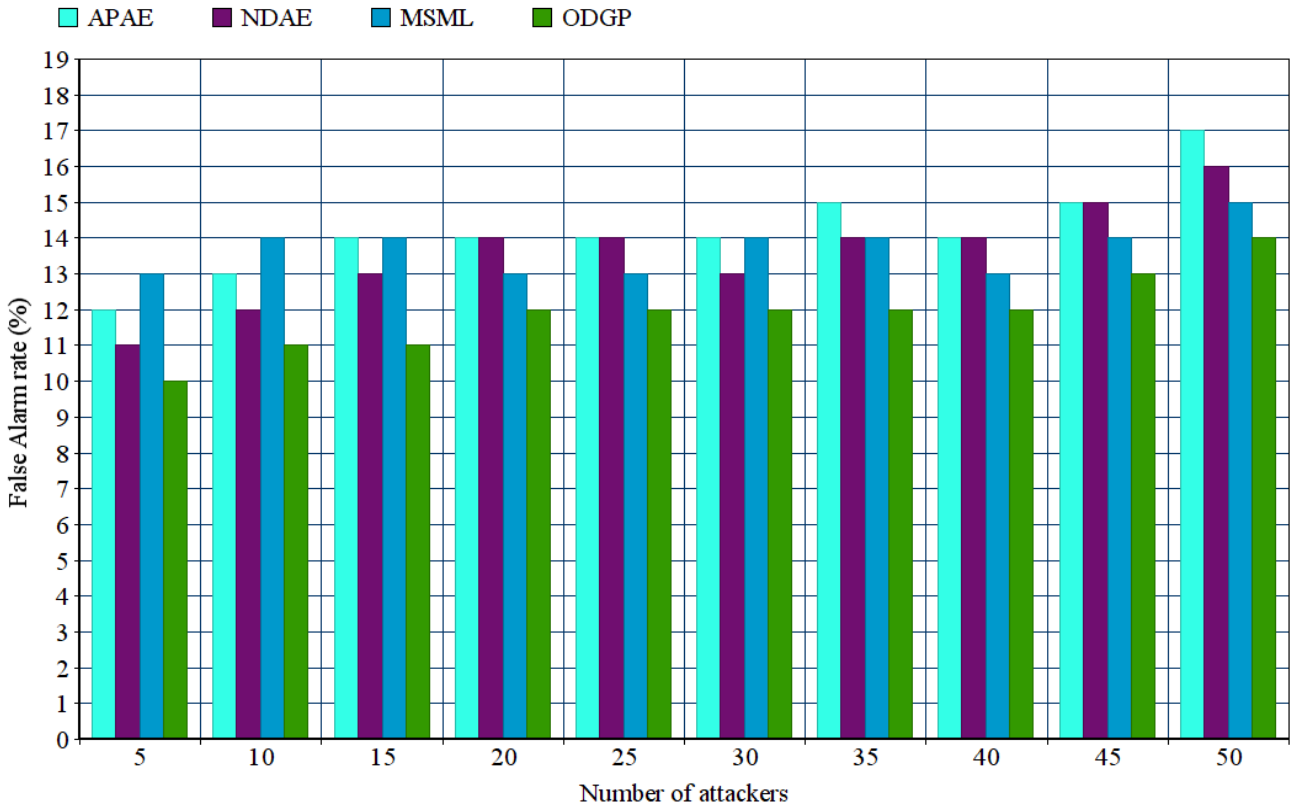

Fig. 8 False alarm rate

Every sensor node trained to detect the intrusion with the subspace set requires storing the detection information. The time consumption incorporates the determination of the subspace of the detector that computes the distance within the detectors in the subspaces. The time complexity for intrusion detection for the proposed technique is $O(n)$, and the complexity is directly proportional to the dimension n. The distance within the detectors in the subspace is denoted as $O(|D|)$. The overall time complexity is computed as $O(n + |D|)$. The total amount of detectors needed to identify the intrusion detection is $\frac{|D|}{(1-\alpha)}$.

5. Conclusion

In this paper, the proposed ODGP technique is one of the intrusions detection-based feature selection techniques; it detects the most common features, which could increase the classification accuracy using a weighted SVM optimizer. The proposed technique has also increased the detection rate with minimized execution time. CICIDS2017 dataset has been utilized to implement the training and testing for the proposed methodology. The performance evaluation is done by comparing the proposed technique with the related techniques of APAE, NDAE, and MSML. In the future, the mathematical prediction methodology with the deep learning technique and the latest feature selection functionality must be included to enhance the detection rate in the network.

References

[1] Abdollahzadeh S, Navimipour NJ, Deployment Strategies in the Wireless Sensor Network: A Comprehensive Review, Comput Commun. 91 (2016) 1–16.

[2] Abualigah LM, Khader AT, Unsupervised Text Feature Selection Technique Based on Hybrid Particle Swarm Optimization Algorithm with Genetic Operators for the Text Clustering, J Supercomput. 73(11) (2017) 4773–4795.

[3] Abualigah L, Diabat A, A Novel Hybrid Antlion Optimization Algorithm for Multi-Objective Task Scheduling Problems in Cloud Computing Environments, Cluster Comput. (2020). https ://doi.org/10.1007/s1058 6-020-03075 -5

[4] Yu Q, Jibin L, Jiang L, An Improved ARIMA-Based Traffic Anomaly Detection Algorithm for Wireless Sensor Networks, Int J Distrib Sens Netw. 12(1) (2016) 9653230.
[5] Rashid B, Rehmani MH, Applications of Wireless Sensor Networks for Urban Areas: A Survey, J Netw Comput Appl. 60 (2016) 192–219.
[6] Pritchard SW, Hancke GP, Abu-Mahfouz AM, Security in Software-Defined Wireless Sensor Networks: Threats, Challenges and Potential Solutions, In: 2017 IEEE 15th International Conference on Industrial Informatics (INDIN), IEEE. (2017) 168–173.
[7] Maza S, Touahria M, Feature Selection for Intrusion Detection Using New Multi-Objective Estimation of Distribution Algorithms, Appl Intell. 49(12) (2019) 4237–4257.
[8] Khasawneh AM, Abualigah L, Al Shinwan M, Void Aware Routing Protocols in Underwater Wireless Sensor Networks: Variants and Challenges. J. Phys Conf Ser. 1550(3) (2020) 032145.
[9] Çavuşoğlu Ü, A New Hybrid Approach for Intrusion Detection Using Machine Learning Methods, Appl Intell. 49(7) (2019) 2735–2761.
[10] Benmessahel I, Xie K, Chellal M, A New Evolutionary Neural Networks Based on Intrusion Detection Systems Using Multiverse Optimization, Appl Intell. 48(8) (2018) 2315–2327.
[11] Aljawarneh S, Aldwairi M, Yassein MB, Anomaly-Based Intrusion Detection System through Feature Selection Analysis and Building Hybrid Efficient Model. J Comput Sci. 25 (2018) 152–160.
[12] Al-Tashi Q, Rais HM, Abdulkadir SJ, Mirjalili S, Alhussian H, A Review of Grey Wolf Optimizer-Based Feature Selection Methods for Classification, In: Evolutionary Machine Learning Techniques, Springer, Singapore. (2020) 273–286.
[13] Abualigah LM, Khader AT, Hanandeh E, Hybrid Clustering Analysis Using Improved Krill Herd Algorithm, Appl Intell. 48(11) (2018) 4047–4071.
[14] Al-Garadi MA, Mohamed A, Al-Ali A, Du X, Ali I, Guizani M, A Survey of Machine and Deep Learning Methods for Internet of Things (Iot) Security, IEEE Commun Surv Tutor. 22 (2020) 1646.
[15] Gao M, Song Y, Xin Y, Intrusion Detection Based on Fusing Deep Neural Networks and Transfer Learning, in Digital TV and Wireless Multimedia Communication: 16th International Forum, IFTC 2019, Shanghai, China, Revised Selected Papers, Springer Nature, Berlin. 1181 (2020) 212.
[16] Rashid A, Siddique MJ, Ahmed SM, Machine and Deep Learning Based Comparative Analysis Using Hybrid Approaches for Intrusion Detection System, In: 3rd International Conference on Advancements in Computational Sciences ICACS: IEEE. (2020) 1–9.
[17] Gamal M, Abbas H, Sadek R, Hybrid Approach for Improving Intrusion Detection Based on Deep Learning and Machine Learning Techniques, Joint European-US Workshop on Applications of Invariance in Computer Vision. Springer. (2020) 225–236.
[18] Araujo-Filho PFd, Kaddoum G, Campelo DR, Santos AG, Maceˆdo D, Zanchettin C, Intrusion Detection for Cyberphysical Systems Using Generative Adversarial Networks in Fog Environment, IEEE Int Things J. (2020) 1–1. Doi: https://doi.org/10.1109/JIOT.2020.3024800.
[19] Li X, Chen W, Zhang Q, Wu L, Building Auto-Encoder Intrusion Detection System Based on Random Forest Feature Selection, Comput Sec. (2020) 101851.
[20] Amir Basati, Mohammad Mehdi Faghih, APAE: An Iot Intrusion Detection System Using Asymmetric Parallel Auto-Encoder, Neural Computing and Applications. (2021). https://doi.org/10.1007/s00521-021-06011-9.
[21] Shone N, Ngoc TN, Phai VD, Shi Q, A Deep Learning Approach to Network Intrusion Detection, IEEE Trans Emerg Topics Comput Intell. 2(1) (2018) 41–50. https://doi.org/10.1109/TETCI.2017.2772792
[22] Yao H, Fu D, Zhang P, Li M, Liu Y, MSML: A Novel Multilevel Semi-Supervised Machine Learning Framework for Intrusion Detection System, IEEE Int Things J. 6(2) (2019) 1949–1959. https://doi.org/10.1109/JIOT.2018.2873125.
[23] (2017). Intrusion Detection Evaluation Dataset (CIC-IDS2017). [Online] Available: https://www.unb.ca/cic/datasets/ids-2017.html