Int J Syst Assur Eng Manag (December 2023) 14(6):2423–2446
https://doi.org/10.1007/s13198-023-02090-0

ORIGINAL ARTICLE

An efficient deep learning-based solution for network intrusion detection in wireless sensor network

Hanjabam Saratchandra Sharma[1] · Arindam Sarkar[2] · Moirangthem Marjit Singh[1]

Received: 17 May 2022 / Revised: 9 June 2023 / Accepted: 1 August 2023 / Published online: 15 August 2023

Abstract This paper introduces a unique intrusion detection method that integrates developmental and operational frameworks, focusing specifically on the wireless sensor network. With the growing number of intrusions, safeguarding sensor nodes has become increasingly crucial. In addition to security breaches, unauthorized access to systems by fraudsters or intruders poses a risk to critical assets. Therefore, detecting and blocking potential threats in the wireless environment is of utmost importance. The proposed detection approach consists of two steps: feature extraction and classification. The study emphasizes the necessity of a distinct intrusion detection method and robust feature extraction and classification techniques. Incorporating a deep learning model is vital for enhancing the precision and accuracy of attack detection. Additionally, it is crucial for efficiency to optimize the CNN architecture's filter size and filter count. The proposed DevOps-based intrusion detection technique involves feature extraction and classification. During the feature extraction stage, statistics and higher-order descriptors are combined with existing characteristics in the early processing of application data. The extracted features are then utilized by the classification method in conjunction with an improved DCNN approach. The technique optimizes the quantity and size of filters in the input vector and fully connected layers. In terms of accuracy as well as FNR, sensitivity, MCC, specificity, FDR, FPR, and NPV, F_1-score against GAF-GYT and other attacks, the suggested technique outperforms conventional models. Specifically, in Application 3, the technique surpasses the DCNN, Innovative Gunner Algorithm, and FAE-GWO-DBN methods by 60.14%, 3.10%, and 5.46%, respectively. Furthermore, for Application 4, the suggested model demonstrates significantly lower FPR rates (91.46%, 67.15%, and 98.4%) compared to the FAE-GWO-DBN, AIG, and DCNN methods. Additionally, the suggested approach outperforms the DCNN, Innovative Gunner Algorithm, and FAE-GWO-DBN approaches by 69.76%, 3.27%, and 22.68%, respectively.

Keywords Wireless sensor network (WSN) · Sensor · Intrusion detection · Feature extraction · Classification · Deep convolutional neural network (DCNN)

1 Introduction

The safety and confidentiality of data must be ensured in the modern world. A variety of fields, including essential infrastructure, healthcare for all, smart cities, autonomous vehicles, etc., benefit from the use of wireless sensor networks (WSN). In the upcoming years, the WSN's utilization will increase dramatically and it will play a significant role in new technical advancements. As a result, the WSN's information safety has expanded along with the network's increasing sensor node count and the amount of information it generates. In WSNs, nodes in the network (sensor nodes) continually acquire perceived information gathered

✉ Arindam Sarkar
arindam.vb@gmail.com

Hanjabam Saratchandra Sharma
sarat.hanjabam@gmail.com

Moirangthem Marjit Singh
marjitm@gmail.com

[1] Department of Computer Science and Engineering, North Eastern Regional Institute of Science and Technology, Nirjuli, Arunachal Pradesh 791109, India

[2] Department of Computer Science and Electronics, Ramakrishna Mission Vidyamandira, Belur Math, Howrah, West Bengal 711202, India

from surroundings and transfer it to the central station via neighboring nodes.

Loss of information while transmission of information is possible as a result of various equipment, network, or attack flaws. More study and research in this field is required to reduce the danger of data loss as a result of security assaults in WSNs. The safeguarding of sensor nodes' activities is necessary for WSN security. Additionally, the vast majority of sensor nodes have network-level connections to outside sources. It is discovered that many WSNs are attackable and are severely harmed. They are significantly impacted because of their lack of capacity to defend. Separately, an attacker will have penetrated the IP layer and gained authority over the WSN node, that the attacker may exploit maliciously. Alternatively, the attacker may have penetrated several security measures in various neighbouring sensor nodes connected to that. vide a broad attack vector, the Mirai botnet (Pour et al. 2020; Koroniotis et al. 2019; Chen et al. 2017) developed a list of gadgets with sensors that were vulnerable. The botnet built up a huge network and was able to generate 600 GB of data every second by installing infected bots, including routers and video cameras. As a result of the assault, several Mirai variants have appeared, taking advantage of sensor nodes' vulnerabilities. Numerous studies on botnet detection have subsequently been reported. Identifying botnets in the WSN while they remain within the targeted node is a challenge for these research. The signature-based strategy and the data analysis-based method are the two main strategies that are commonly used to address issues in the present research. The signature-based strategy produces complexity since the abnormalities and attacks have been documented in the database (Alauthman et al. 2020; Asadi et al. 2020; Mousavi et al. 2020; Jung et al. 2020). Specialists from a variety of disciplines (Giridhar Reddy and Sai Ambati 2020) are paying close focus on computational intelligence techniques. These methods, however, need a substantial amount of tagged cases. Therefore, more research in this area is required for the most accurate identification of WSN threats. It can be susceptible to attack as a consequence. The operational method is quicker when employing an analysis of data technique than it is with existing ways, and the issue of unanticipated dangers is easily handled. Additionally, a number of machine learning methods (Azar et al. 2019; Alfan et al. 2020; Shafq et al. 2020; Cheng et al. 2020), both supervised and unsupervised, are being used to improve the accuracy of WSN detection of attacks. The labelled data is used by the machine learning with supervision methods, and each instance has a label that describes a certain sort of assault. To detect WSN attacks, supervised models for machine learning such as neural networks, K-nearest neighbour, deep learning, and support vector machines were utilized.

The primary difficulties are:

1. A unique intrusion detection method, as well as robust feature extraction and classification approaches, are required.
2. To improve the identification of attacks precision and accuracy, a model that uses deep learning is needed.

Table 1 Contemporary techniques: characteristics and shortcomings

Author	Characteristics	Shortcomings
Shailendra Rathore and Park (2018)	Enhances the effectiveness of detection. A higher rate of precision	When confronted with a problem that isn't well-posed. Possibility of reaching a lower level of success
Murali and Jamalipour (2020)	Excellent accuracy. A greater favourable result percentage is also attained	To create a trustworthy identification system, further investigation is required
Nguyen et al. (2020)	A smaller amount of space is needed, and analysis takes a shorter amount of time. increases the percentage of genuine positives	It is necessary to prevent an attack using brute force throughout the entire PSI graph
Jung et al. (2020)	Improved F1-measure improvement is more precise	Information on consumption of energy is analysed using a deep neural network classifier
Baig et al. (2020)	Improved precision The period between arrivals is greater	The effectiveness of identifying still has space for improvement. Numerous variables demand a careful analysis
Liu et al. (1989)	More accurate detection Increased precision in prediction	Different techniques for conducting various assaults collaboratively are in need of examination
Hasan et al. (2019)	IoT system intrusions were successfully stopped. More occurrences can be predicted with better precision	As there are more concerns, the strong detection mechanism has to be improved
Ho (2018)	Develops the capacity to identify risks in the midst of difficulty. The proper sequence of detection	It only takes some tests. It has substantial costs

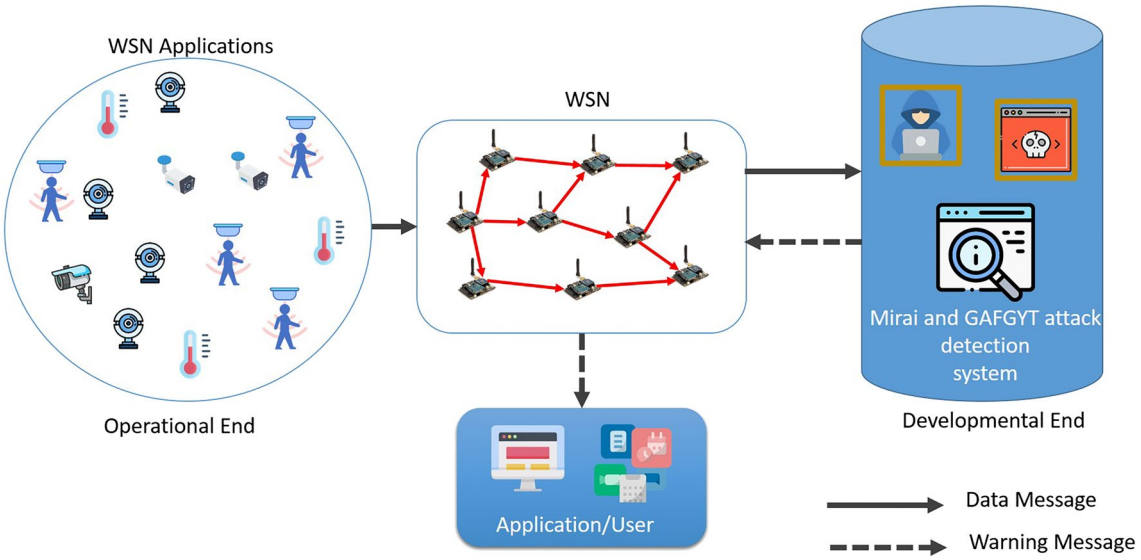

Fig. 1 Proposed threat detection approach

3. An efficient approach is required to optimize the number of filters and their size in CNN.

This method has several objectives, listed as follows:

1. Analyzing information from each app by combining statistical and advanced statistical characteristics with existing features during the feature extraction step.
2. A deep convolutional neural network (DCNN) model is used to develop the classification framework that focuses on the retrieved characteristics.
3. The effectiveness of the recommended approach is tested through tests, and the results show that it functions better than comparable methods already in use.

The various contributions provided by this article are outlined as follows:

1. A novel intrusion detection approach is as developed by connecting the DevOps architecture with two steps: feature extraction and classification. Each application's data was processed early in the feature extraction process by integrating statistics and higher-order descriptors with the current features.
2. The classification algorithm is developed employing these extracted features using an enhanced DCNN technique.
3. A novel approach is employed to minimize the number of filters and filter size in both the fully connected layers and the input vector.
4. In regards to sensitivity, accuracy, and specificity, as well as TPR (True_Positive_Rate), TNR (True_Negative_Rate), PPV (Positive_Precdictive_Value), NPV (Negative_Predictive_Value), FPR (False_Positive_Rate), FNR (False_Negative_Rate), FDR (False_Discovery_Rate), MCC (Mathews_Correlation_Coefficient), and F1-score under the GAF-GYT and Mirai attacks the suggested work does better than other standard approaches.
5. In application 3, the proposed methodology beats the DCNN, Innovative Gunner Algorithm, and FAE-GWO-DBN (Pijarski and Kacejko 2019) approaches by 60.14 %, 3.10 %, and 5.46 %, respectively. Furthermore, the recommended approach for application 4 has a low FPR, that is superior by 91.46 %, 67.15 %, and 98.4 %, respectively, than FAE-GWO-DBN, AIG, and DCNN approaches. The suggested strategy also beats the DCNN, Innovative Gunner Algorithm, and FAE-GWO-DBN techniques by 69.76 %, 3.27 %, and 22.68 %, respectively.

Related work is explained in Sect. 2. Section 3 presents the suggested technique. Section 4 presents the enhanced optimisation for resolving optimization problems. The evaluation and outcomes are covered in Sect. 5. The findings and

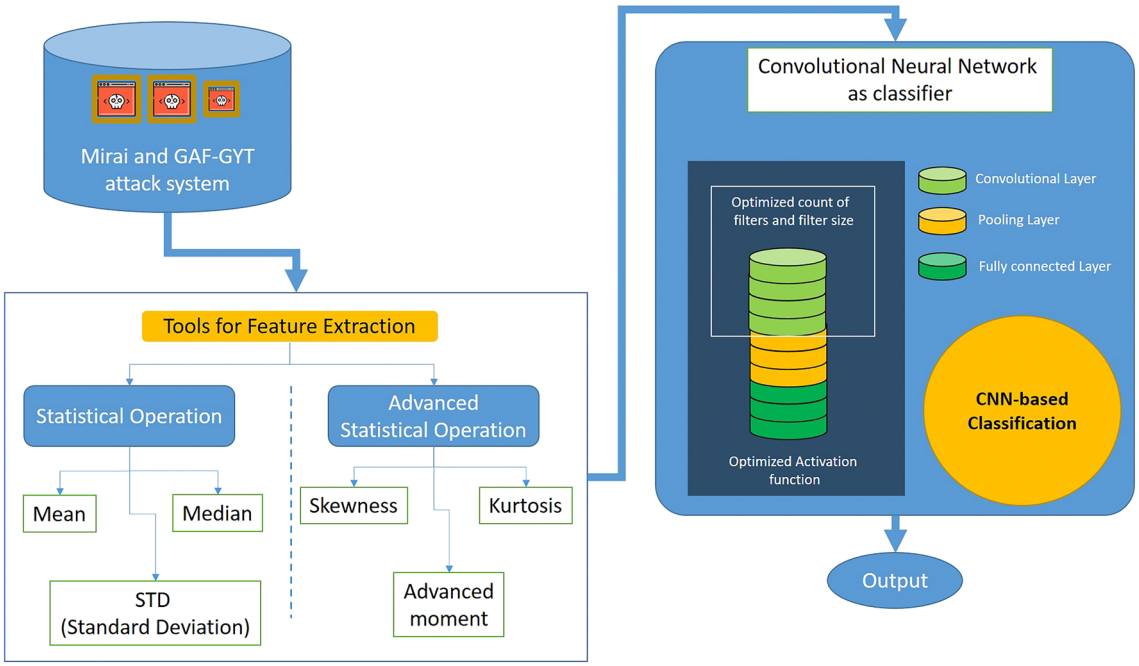

Fig. 2 The concept of the suggested threat detection technique in WSN

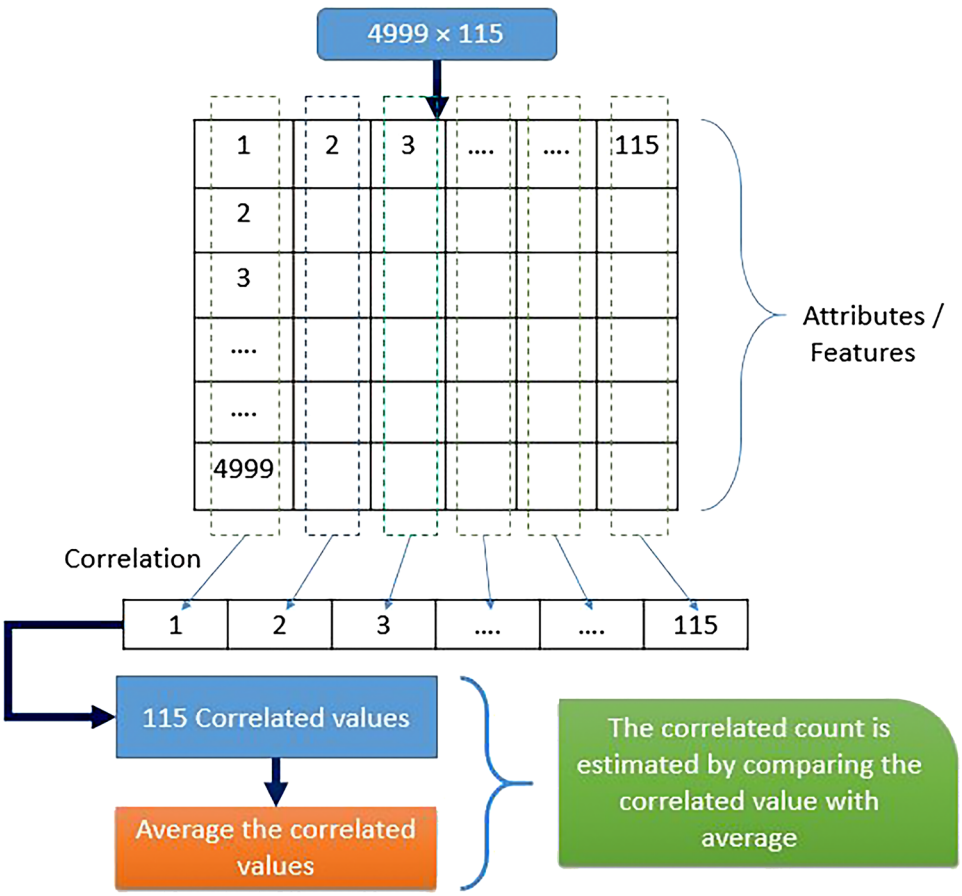

Fig. 3 A model of the recommended extraction of features in action

future scope are included in Sect. 6, along with references at the end.

2 Related Work

Klassen and Yang (202) proposed an anomaly-based intrusion detection employing the Bayesian classifier in WSN. They investigated an Adhoc network with three types of attacks i.e. a DoS attack, black hole attack, and malicious attack to study if any harmful activities can be detected in time. A network having 33 numbers of nodes following AODV was built and collected the traffic data. Singh and Singh (2017) offered an AHIDS (advanced hybrid intrusion detection system) using a multilayered perceptron NN (neural network) containing the supervised learning network's feed forward neural networks and backpropagation neural network based on the fuzzy logic mechanism. The suggested mechanism identifies and defends wormhole and Sybil assaults in WSN against hello flooding. Shaon and Ferens (2015), proposed a technique for the detection of wormhole intrusions in WSN utilizing an Artificial Neural Network (ANN).

The suggested work's primary goal is to identify wormhole assaults in both uniform and non-uniform environments. Singh et al. (2020)demonstrated a method for detecting wormhole attacks in WSNs using ANN. Sherazi et al. (2019) addressed Intrusion Prevention System (IPS)-based protection and recommended a Q-learning and fuzzy logic strategy. The investigation was conducted using a tuple of four parameters as its foundation. On a 6BR machine that continuously evaluates internet packets, the suggested technique included techniques for Q-Learning and Fuzzy Logic.

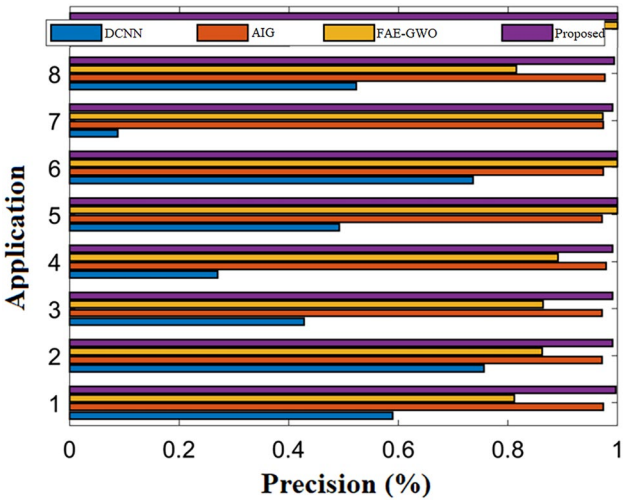

Fig. 5 Precision comparisons in terms of positive measure for Mirai attack

They noticed that DDoS-induced communication bottleneck was caused by packets flooding. Mourabit et al. (2015) used Random Tree, NaiveBayes, K-means, and Support Vector Machine algorithms to recognize various forms of attacks, including spoofed, changed, or replayed routing data attack, Picked forwarding attack, sinkhole attack, tampering, jamming, Sybil, Hello floods, and spoofing of acknowledgement. Sandhya and Julian (2014) proposed an IDS (intrusion detection system) by using K-means. The end result was an elevated probability of identification and a low incidence of false alarms. The proposed system using K-means proved to be suitable for dynamic environments. The system intelligently analyzed the generated intrusion alerts and new attacks are also detected that lacks intrusion signature on the

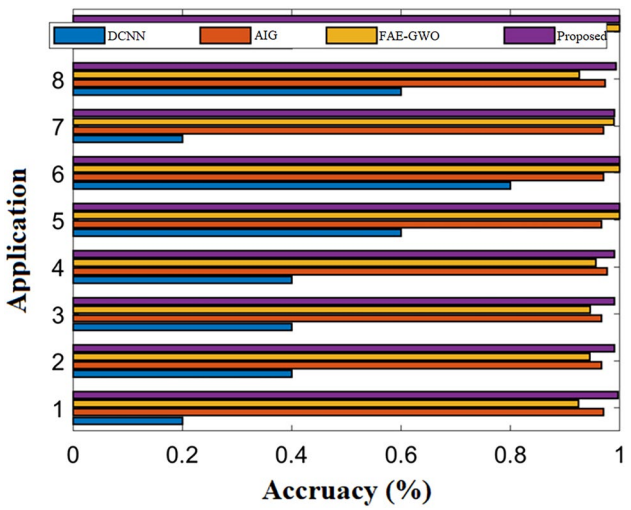

Fig. 4 Accuracy comparisons in terms of positive measure for Mirai attack

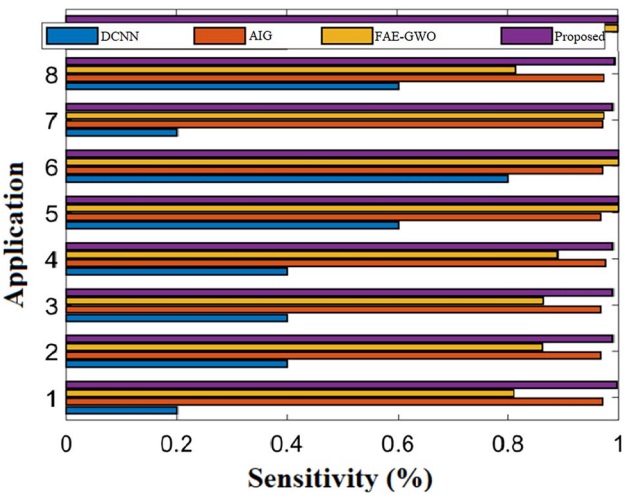

Fig. 6 Sensitivity comparisons in terms of positive measure for Mirai attack

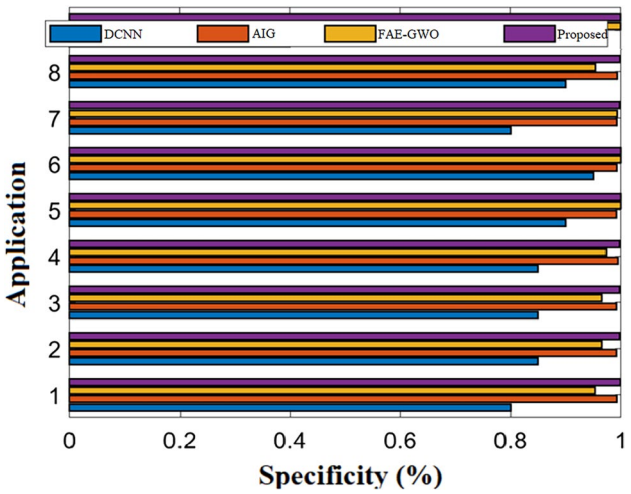

Fig. 7 Specificity comparisons in terms of positive measure for Mirai attack

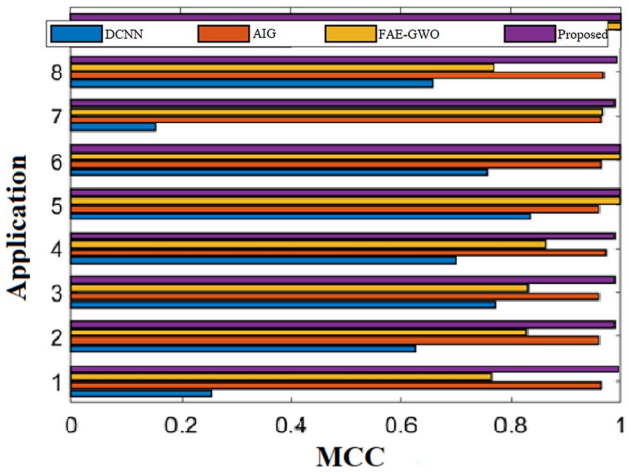

Fig. 9 MCC comparisons in terms of positive measure for Mirai attack

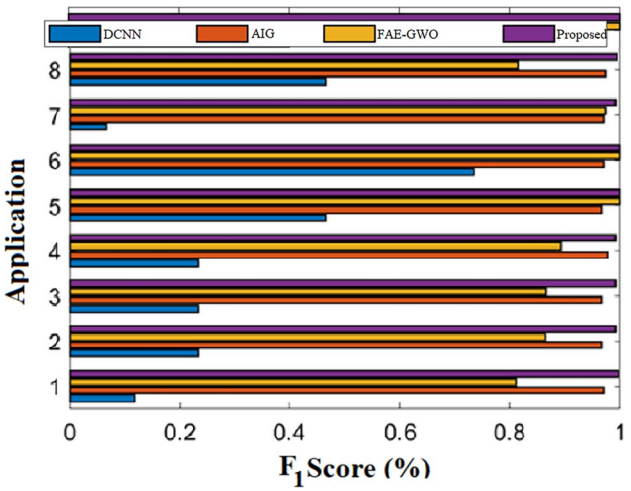

Fig. 8 F_1-score comparisons in terms of positive measure for Mirai attack

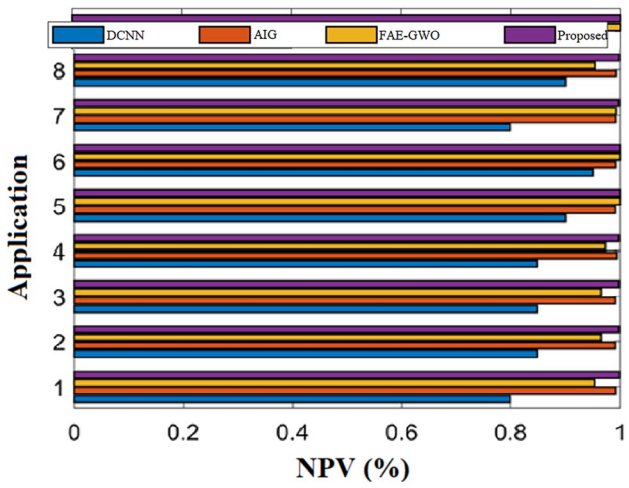

Fig. 10 NPV comparisons in terms of positive measure for Mirai attack

basis of genetic K-means algorithm. Maleh et al. (2015) proposed an SVM (Support vector machines) based hybrid IDS (intrusion detection system) for wireless sensor network. A detection technique and a learning algorithm was used based on SVM to identify intrusion based on the signatures of the attack. Ho (2018) created a methodology in 2018 that combines probabilistic assessments and SPRT packets put into industrial IoT devices to effectively and reliably discover code-reuse concerns. The suggested attack detection method was evaluated and tested in commercial Internet-of-Thing devices. Numerous tests have revealed that the suggested approach has produced averaged detection precision for both a large and small collection of coding reused packets. In 2018, Shailendra Rathore and Park (2018) introduced an attack detection technique that utilizes fog computing that relies on a recently developed ESFCM framework and fog computing hypothesis. Semi-supervised fuzzy c-means has been employed in both the ESFCM approach for processing labelled data and an ELM strategy for improving the accuracy of classification in a more rapid detection rate. The created model outperforms the centralized intrusion detection process, according to the computations using the NSLKDD database. In specifically, the devised approach attained an identification time of 11 milliseconds and an accuracy of 86.53%. To precisely recognise anomalies and attacks in IoT gadgets, Hasan et al. (2019) focused on analyzing the outcomes across multiple ML approaches. ML techniques employed in the present research were Decision Trees (DT), Linear Regression, Artificial Neural Network (ANN), Support Vector Machines, and Random Forest (RF). Note that

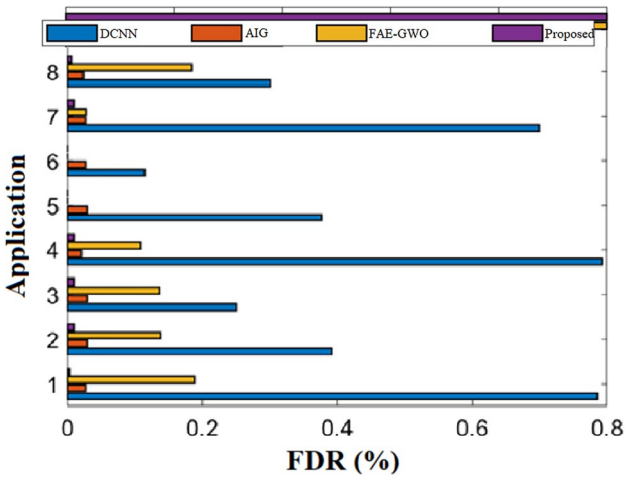

Fig. 11 FDR comparisons in terms of negative measure for Mirai attack

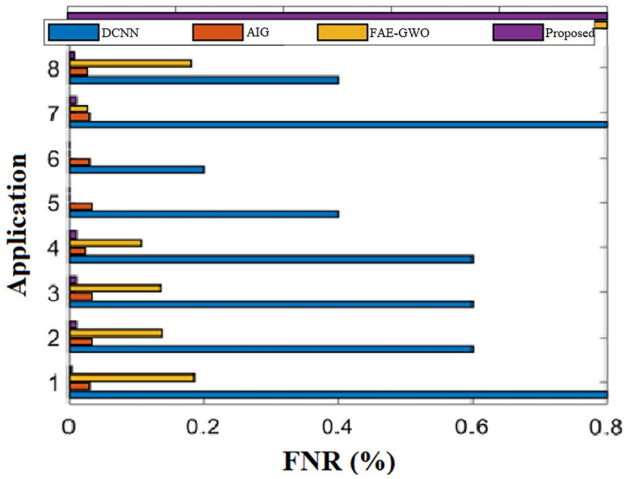

Fig. 12 FNR comparisons in terms of negative measure for Mirai attack

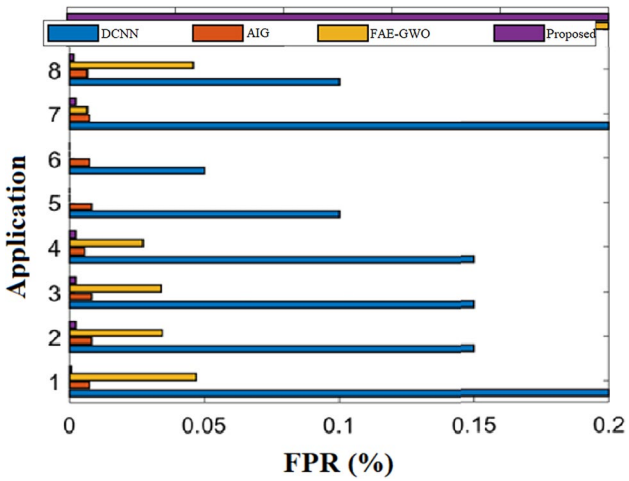

Fig. 13 FPR comparisons in terms of negative measure for Mirai attack

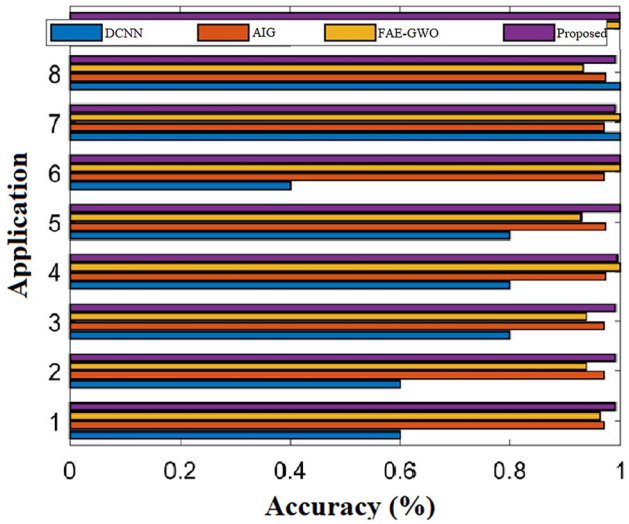

Fig. 14 Accuracy comparisons in terms of positive measure for GAF-GYT attack

the study's results have been distinguished using precision, F1-score, and area beneath the Receiver Operating Characteristic Curve. A 99.4% accuracy rate was attained for DT, RF, and ANN. Overall, the analysis shows that Random forest works better than other classifiers. In 2019, Liu et al. (1989) introduced the idea of a "multiple-mix-attack approach." Then, the PD prototype perceptron and K-means approach was developed for recognizing intruders and determining the level of confidence in sensor nodes. Employing PDE, an updated perceptron modelling learning technique, the identification rate was increased. The network route was made better to do this. The exploratory investigation showed that PDE and PD had superior detection of dangerous nodes in comparison to other similar algorithms with more accuracy rates. In 2020, Baig et al. (2020) suggested a denial-of-service (DoS) assault strategy that involved sending a large number of network packets to a specific set of network node sensors. This denial-of-service assault has the potential to impair normal operations and result in devastating losses for emergency services. As a part of this experiment, an intelligent DoS detection strategy has been created, which includes components for feature ranking and creation, testing and training, and data production. For this suggested framework, an experimental evaluation was conducted using real-world IoT threat scenarios. As a consequence, the applied work has obtained higher accuracy as compared to classification techniques. To protect the health sector from harmful cyberattacks, Jung et al. (2020) plan to categorize IoT devices that are influenced by malevolent activities

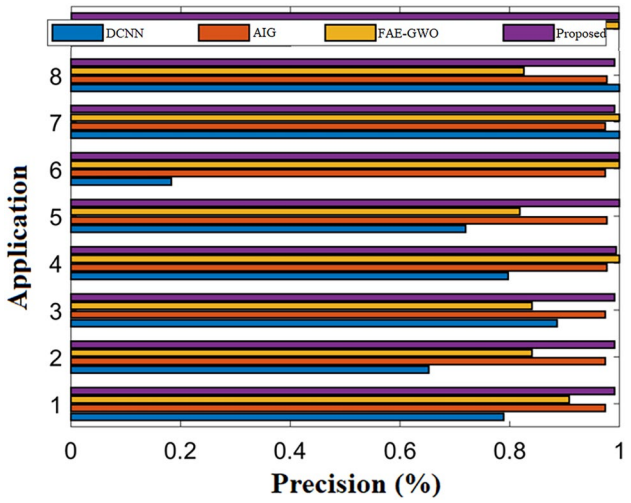

Fig. 15 Precision comparisons in terms of positive measure for GAF-GYT attack

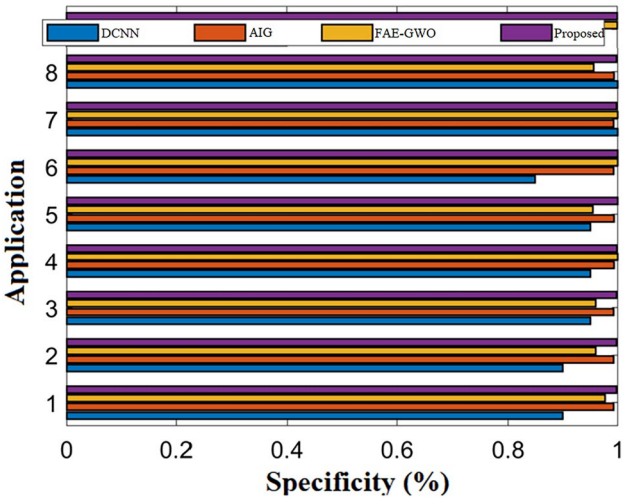

Fig. 17 Specificity comparisons in terms of positive measure for GAF-GYT attack

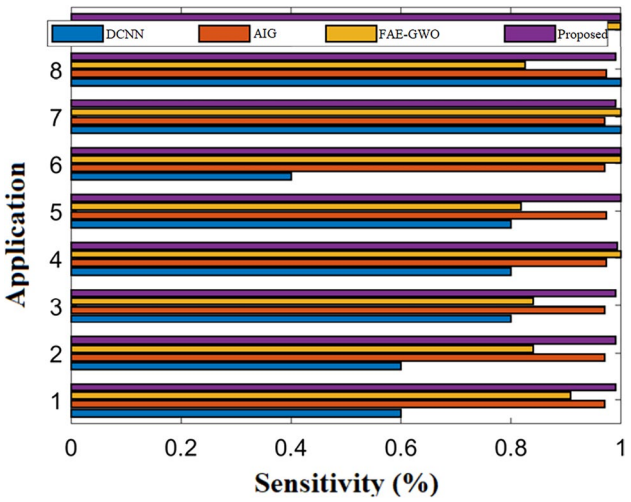

Fig. 16 Sensitivity comparisons in terms of positive measure for GAF-GYT attack

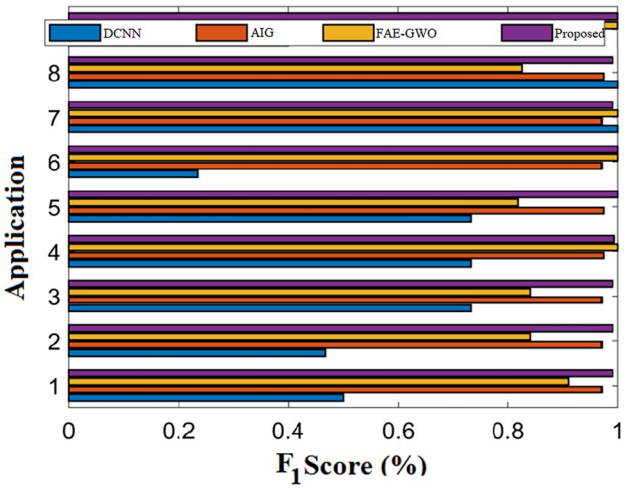

Fig. 18 F_1 comparisons in terms of positive measure for GAF-GYT attack

based on power consumption patterns in 2020. A CNN-based deep learning method, consisting of an eight layer convolutional neural network and a unit for processing of data, has been built for this goal. To help the CNN in achieving better precision, the data was segmented and normalized before it was deployed. The efficiency was calculated by running cross-device assessments, leave-one-botnet-out assessments, self-evaluation, and leave-one-device-out assessments on three common Internet-of-Things device types: routers, digital assistant systems, and security cameras, and the results showed that the efficiency seemed to be better in the accuracy rate. Nguyen et al. (2020) contributed several advances to IoT intrusion detection in 2020. A PSI-rooted functionality based on subgraphs was generally supplied to identify DDoS assaults. Second, a limited set of attributes with precise behavioral descriptors were created, requiring less processing time and less storage capacity. The resilience and efficiency of suggested characteristics over five machine learning classifiers were therefore justified by the study. As a result, each classifier does have a good suggestion with little processing time and a higher identification rate than existing techniques. In order to ascertain the Sybil assault, Murali and Jamalipour (2020) have developed an ABC-motivated, dynamic assault modeling, and a portable RPL compact intrusion prevention system. In addition, depending on their actions, three different classifications of the Sybil

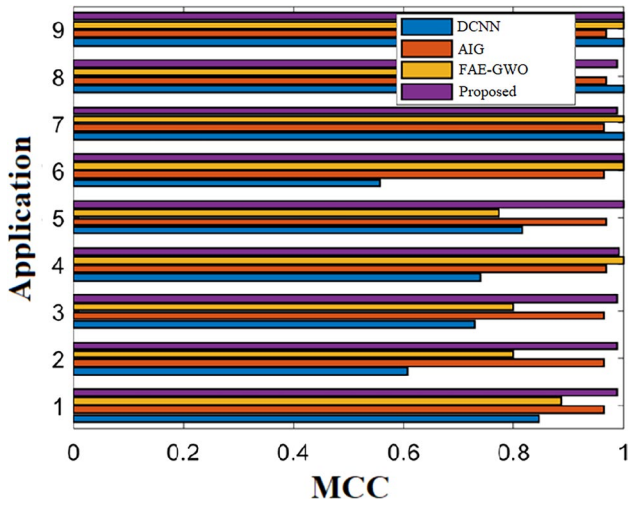

Fig. 19 MCC comparisons in terms of positive measure for GAF-GYT attack

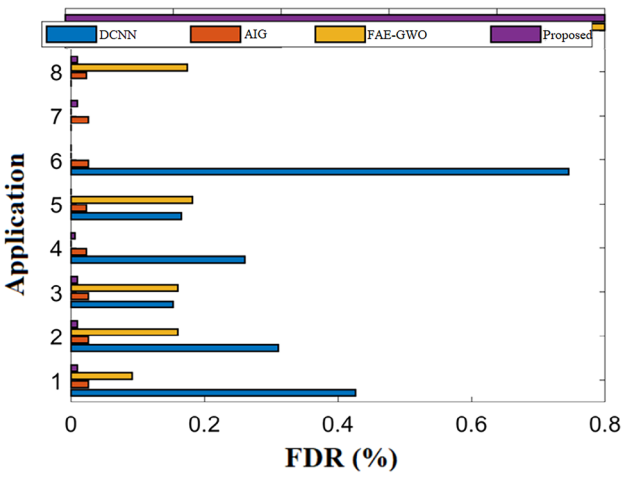

Fig. 21 FDR comparisons in terms of negative measure for GAF-GYT attack

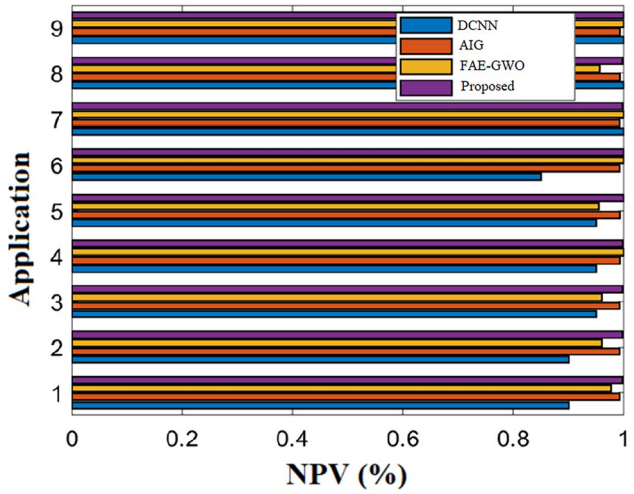

Fig. 20 NPV comparisons in terms of positive measure for GAF-GYT attack

assault were explored. Furthermore, under this Sybil assault, the RPL efficiency was examined in terms of traffic overlay management, energy usage, and packet delivery ratio. Furthermore, the suggested study was evaluated in terms of sensitivities, precision, and specificity measurements.

The distinctive characteristics and difficulties of the most advanced techniques are highlighted in Table 1 below.

3 Proposed methodology

This article presents its meaningful impact on DevOps and proposes a unique concept for ensuring security using a threat detection system. The general idea of DevOps is depicted in Fig. 1. In this proposed threat detection approach, the DevOps architecture covers both development and operations. The developmental scenario is used in the development stage, whereas the operational scenario is used in the operating section (apps). This development end handles the entire work of application security assurance, which is made possible by calculating all applications' data. The presented WSN intrusion detection procedure manages data security-related assurance, allowing the assaults in WSN to be identified and warnings to be provided to the appropriate applications. In the next section, various steps for detecting an assault in WSN are given.

The major purpose of this analysis is on detecting WSN assaults, in which a unique intrusion checking approach comprising two steps is developed: extraction of features and classifying them. The information analysis is the first step, and it is taken from a database (archiveicsuciedu 2021) with following seven apps.

- App1: Samsung_SNH_1011_N_Webcam
- App2: Danmini_Doorbell
- App3: Ecobee_Thermostat
- App4: Ennio_Doorbell
- App5: Philips_B120N10_Baby_Monitor
- App6: Provision_PT_737E_Security_Camera
- App7: SimpleHome_XCS7_1002_WHT_Security_Camera

Those acquired data $E = \{e_1, e_2, ..., e_\alpha\} E_{\alpha\times\beta} = \begin{Bmatrix} e_{11} & e_{12} & ....e_{1\beta} \\ e_{21} & e_{22} & ....e_{2\beta} \\ .... & .... & ..... \\ e_{\alpha 12} & e_{\alpha 2} & ....e_{\alpha\beta} \end{Bmatrix}$ from various applications are then

Table 2 TPR comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.998446535	0.998820357	0.999153578	0.996040351	0.999405672	0.996314496	0.999260492
MEDIAN	0.999913696	0.999157398	0.998495251	0.996323183	0.999915096	0.9995086	0.999907561
STD	0.996288945	0.999325918	0.999529766	0.995474687	0.999405672	0.996232596	0.997134406
KURTOSIS	1	0.99991574	0.999905953	0.999717168	0.999915096	0.9996724	0.999907561
SKEWNESS	0.999827393	0.99991574	0.999811906	0.999622891	0.999830192	0.9995086	0.999815123

Table 3 TNR comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.99970176	0.999877376	1	0.998869737	0.999878349	0.995945946	1
MEDIAN	0.999900587	0.999877376	1	0.999576151	0.999969587	1	1
STD	0.999304106	0.999877376	1	0.999293586	0.999908762	0.998918919	0.999069335
KURTOSIS	1	1	1	1	1	1	0.999883667
SKEWNESS	1	1	1	1	0.999969587	1	1

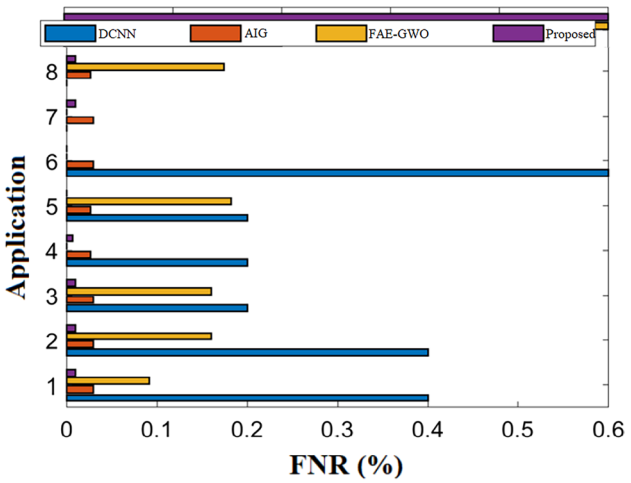

Fig. 22 FNR comparisons in terms of negative measure for GAFGYT attack

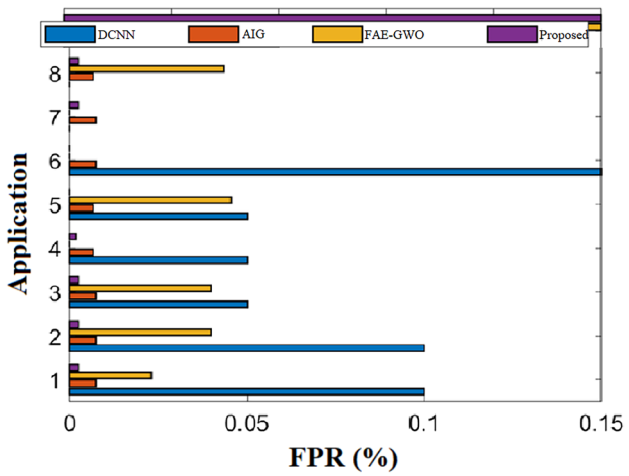

Fig. 23 FPR comparisons in terms of negative measure for GAFGYT attack

subjected to pre-processing, wherein the normalizing is assessed to handle the data within the range of 0 to 1. This is then safely stored for further use. The following section depicts the normalizing procedure.

Database normalization is the process of organizing information in a database and has been performed even before extracting features. The is described in Eq. 1.

$$\delta = \frac{\widehat{e_{jk}}}{\underset{j=1\ to\ \gamma,\ k=1\ to\ \delta}{max(\widehat{e_{jk}})}} \tag{1}$$

The analytical and higher-order statistical characteristics are retrieved from all these normalized data $Y = \{y_1, y_2, ..., y_{\beta}\}$ during the feature extraction stage. $Gu_1 = g_1, g_2, g_3$ refers to statistical characteristics like average, median, and standard deviation, whereas$Gu_2 = i_1, i_2, i_3$ refers to advance statistical characteristics like kurtosis, skewness, and relatively higher-order moments (Sarma 2021; Sharma et al. 2023). Following that, those characteristics are concatenated with the normalized data $Gu = [Y\ Gu_1\ Gu_2]$, and features are extracted are produced. The categorization process is subsequently carried out with CNN's assistance. This study employs an optimal situation in which the number of filters, as well as the size of a filter in the convolution layers and the activation function, are ideally optimized to maintain an effective detection performance. The entire concept of the suggested threat detection technique in WSN is shown in Fig. 2.

Table 4 PPV comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.999740754	0.999915647	1	0.999243356	0.999660297	0.996314496	1
MEDIAN	0.999913696	0.999915676	1	0.999716205	0.999915096	1	1
STD	0.999393992	0.99991569	1	0.999526694	0.999745201	0.999014455	0.999258916
KURTOSIS	1	1	1	1	1	1	0.999907561
SKEWNESS	1	1	1	1	0.999915089	1	1

Table 5 NPV comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.998213222	0.998285994	0.996539792	0.994094488	0.999787131	0.995945946	0.9990702
MEDIAN	0.999900587	0.99877511	0.993865031	0.994517852	0.999969587	0.999459751	0.99988368
STD	0.995740466	0.999019848	0.998074702	0.993259374	0.999787137	0.995868511	0.996403295
KURTOSIS	1	0.999877391	0.999614346	0.999576331	0.999969588	0.999639769	0.999883667
SKEWNESS	0.999801213	0.999877391	0.99922899	0.999435188	0.999939176	0.999459751	0.999767388

Table 6 FPR comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.00029824	0.000122624	0	0.001130263	0.000121651	0.004054054	0
MEDIAN	9.94E−05	1.23E−04	0.00E+00	4.24E−04	3.04E−05	0.00E+00	0.00E+00
STD	6.96E−04	1.23E−04	0.00E+00	7.06E−04	9.12E−05	1.08E−03	9.31E−04
KURTOSIS	0	0	0	0	0	0	0.000116333
SKEWNESS	0	0	0	0	3.04127E−05	0	0

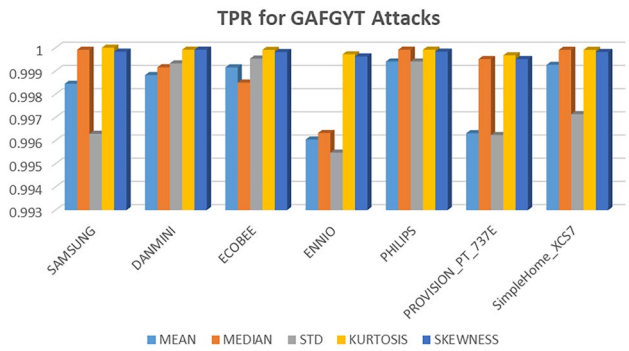

Fig. 24 TPR comparisons graph of GAFGYT attack

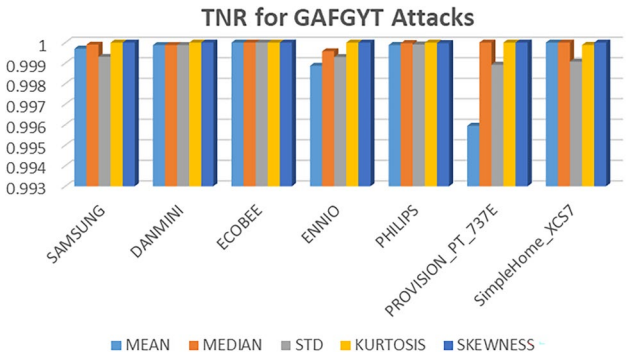

Fig. 25 TNR comparisons graph of GAFGYT attack

The statistics that have been incorporated already comprise the current characteristics, as well as statistical and advanced statistical features that have been combined. Mean, median, and standard deviation are statistical properties, whereas higher-order moments, kurtosis, and skewness are advance statistical characteristics. These labels or characteristics were subjected to correlation, resulting in the associated values. Thereafter, the associated data are averaging and evaluated to the precise mean value. As a consequence, the counts of related values with a larger or comparable mean value is recorded. Figure 3 shows a model of the recommended extraction of features in operation.

The square root of the variance Y is the standard deviation μ, which is given in Eq. 2.

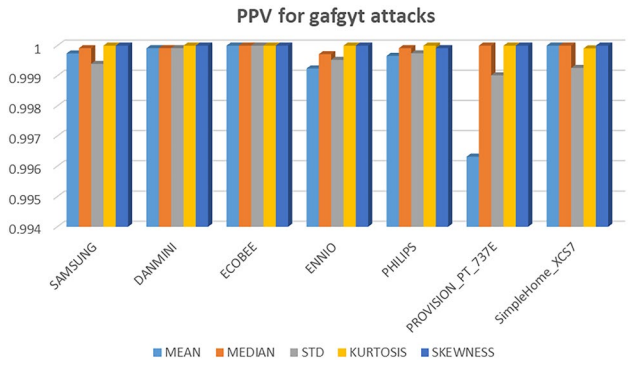

Fig. 26 PPV comparisons graph of GAFGYT attack

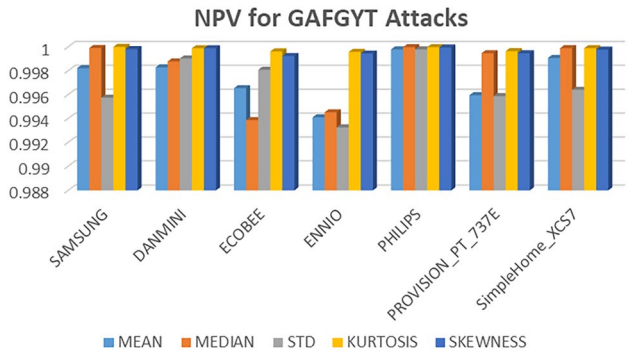

Fig. 27 NPV comparisons graph of GAFGYT attack

$$\mu = \sqrt{F\left[(Y-\eta)^2\right]} \tag{2}$$

The variance of an arbitrary parameter Y is the acceptable magnitude of the squared deviation from the average Y,$\eta = F(Y)$, as is shown in Eq. 3.

$$Var(Y) = F\left[(Y-\eta)^2\right] \tag{3}$$

The arithmetic averages are calculated by adding the magnitude of each sample with the available number of samples. The assessment of average is performed utilizing Eq. 4 on a sample including collected data $y_1, y_2, ..., y_{\ddot{n}}$ entries.

$$mean(\eta) = \frac{1}{\ddot{\beta}} \sum_{j=1}^{\ddot{\beta}} y_j, \tag{4}$$

The moment is a numerical measure of a function's form. On the basis of Eq. 5, the $\ddot{\beta}$-th instant of a function $f\left(\hat{y}\right)$ of an real variable $\ddot{d}$ is given.

$$\eta_{\ddot{\beta}} = \int_{-\infty}^{\infty} (y-\ddot{d})^{\ddot{\beta}} f(y) dy \tag{5}$$

The merged information *Gu* would then be submitted to classification using the characteristics generated during in the feature extraction process.

The "tailedness" of the likelihood distribution for a real-valued randomised vector is measured by kurtosis. This is stated with the abbreviation Eq. 6.

$$Kurt(Y) = F\left[\left(\frac{(Y-\eta)}{\mu}\right)^4\right] \tag{6}$$

Skewness is a measure of asymmetrical probability distribution with a true random vector. Based on Eq. 7, the skewness π_1 of the random vector Y is determined.

Table 7 FNR comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.001553465	0.001179643	0.000846422	0.003959649	0.000594328	0.003685504	0.000739508
MEDIAN	8.63E−05	8.43E−04	1.50E−03	3.68E−03	8.49E−05	4.91E−04	9.24E−05
STD	0.003711055	0.000674082	0.000470234	0.004525313	0.000594328	0.003767404	0.002865594
KURTOSIS	0	8.42602E−05	9.40468E−05	0.000282832	8.49041E−05	0.0003276	9.24385E−05
SKEWNESS	0.000172607	8.42602E−05	0.000188094	0.000377109	0.000169808	0.0004914	0.000184877

Table 8 FDR comparisons of GAFGYT attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.996314496	8.43526E−05	0	0.000756644	0.000339703	0.003685504	0
MEDIAN	8.63E−05	8.43E−05	0.00E+00	2.84E−04	8.49E−05	0.00E+00	0.00E+00
STD	0.000606008	8.43099E−05	0	0.000473306	0.000254799	0.000985545	0.000741084
KURTOSIS	0	0	0	0	0	0	9.24385E−05
SKEWNESS	0	0	0	0	8.49113E−05	0	0

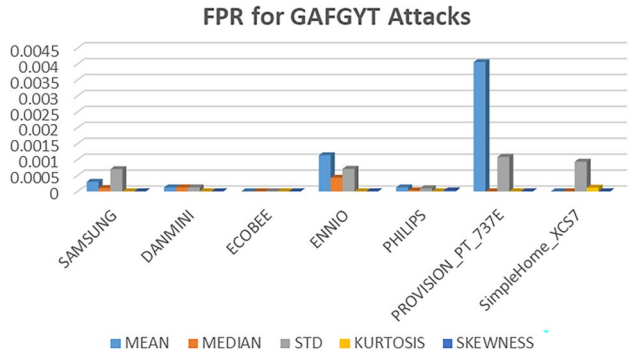

Fig. 28 FPR comparisons graph of GAFGYT attack

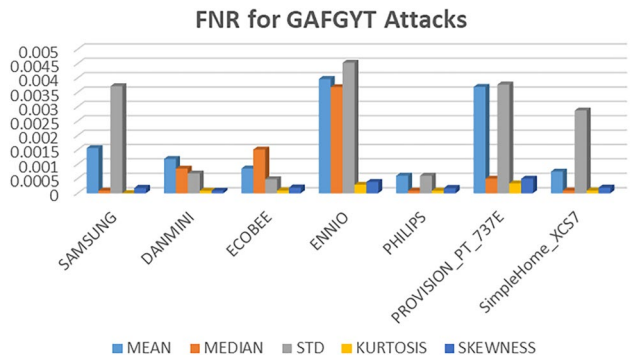

Fig. 29 FNR comparisons graph of GAFGYT attack

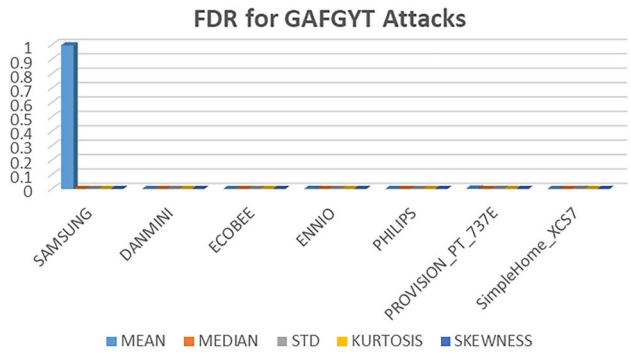

Fig. 30 FDR comparisons graph of GAFGYT attack

$$\pi_1 = F\left[\left(\frac{(Y-\eta)}{\mu}\right)^3\right] \tag{7}$$

Though after incorporating machine learning tasks into NNs, previous knowledge integration into the network design is critical for excellent generalization performance. Convolutional neural network achieve its fundamental goal of spatial information practice.

The convolution layers must employ tiny filters Q_t (e.g. 3x3 to the maximum as 5x5), depending on a $Stride = 1$, and filling the input vector using 0 s, despite the fully connected layers not changing the given spatial size of the input. The suggested method is used to optimize the filter length R_T as well as the amount of filters R_O in this paper.

Suppose that the Fully connected layer is *dm*. As a result, the layer $bk'u$ input comprises $q_1^{(dm-1)}$ extracted features from the previous layers, each with a size of $q_2^{(dm-1)} \times q_3^{(dm-1)}$. Even when$dm = 1$, the source remains the only information *dm*, that is made up of one or even more streams, and which receives raw information as input to convolutional neural network. The result of the layer *dm* comprises q_1^{dm} characteristics maps of length $q_2^{dm} \times q_3^{dm}.\hat{Y}_j^{dm}$. The *j*-th characteristics maps in layer *dm* is delineated by $\hat{Y}_j^{dm}$ which is defined according to Eq. 8.

$$\hat{Y}_j^{dm} = E_j^{(dm)} + \sum_{k=1}^{q_1^{(dm-1)}} Q_{j.k}^{(dm)} * \hat{Y}_k^{(dm-1)} \tag{8}$$

Where $E_j^{(dm)}$ represents the biased 2D array, and $Q_{j.k}^{(dm)}$represents the filter of length $2t_1^{dm} + 1 \times 2s_2^{dm} + 1$ coupling the k^{th} characteristics map in a layer $dm - 1$ with the characteristics map in *dm*. The length of the result characteristics graph was determined using Eq. 9.

$$q_2^{dm} = q_2^{(dm-1)} - 2t_1^{dm} and\ q_3^{dm} = q_3^{(dm-1)} - 2t_2^{dm} \tag{9}$$

$Q_{j.k}^{(dm)} = Q_{j.l}^{(dm)}$ as $k \neq l$ are repeatedly used to measure the uniqueness of the fixed characteristic map for $k = l$. All characteristics map $\hat{Y}_j^{dm}$ in the layer *dm* is made up of matrix of $q_2^{dm}.q_3^{dm}$ components. Eqs. 10 and 11 show how to determine the result based upon on the component at location (h, i).

Table 9 TPR comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0	0.999263984	1	0	1	0.999833569	0.996838261
MEDIAN	0	0.999165849	0.999381161	0	1	0.999916785	0.999732811
STD	0	0.99936212	1	0	1	0.997004244	0.751380477
KURTOSIS	0	0.999901865	1	0	1	0.999833569	1
SKEWNESS	0	0.999901865	1	0	1	0.999916785	1

Table 10 TNR comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0	0.998279676	0.998118178	0	1	0.99973041	0.997856377
MEDIAN	0	0.999754239	0.998870907	0	1	0.99946082	0.99964273
STD	0	0.995207668	0.999247271	0	0.99996956	0.998921639	0.97653924
KURTOSIS	0	1	1	0	1	1	0.99988091
SKEWNESS	0	1	1	0	1	1	1

Table 11 PPV comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0	0.999313018	0.999779035	0	1	0.999750374	0.999196536
MEDIAN	0	0.999901792	0.999867327	0	1	0.999500915	0.999866388
STD	0	0.998088797	0.999911602	0	0.999945658	0.998999416	0.988459285
KURTOSIS	0	1	1	0	1	1	0.99995547
SKEWNESS	0	1	1	0	1	1	1

Table 12 NPV comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0	0.998157022	1	0	1	0.999820257	0.991597633
MEDIAN	0	0.997914878	0.994752624	0	1	0.999910096	0.999285714
STD	0	0.998397436	1	0	1	0.996771879	0.59493579
KURTOSIS	0	0.9997543	1	0	1	0.999820305	1
SKEWNESS	0	0.9997543	1	0	1	0.999910145	1

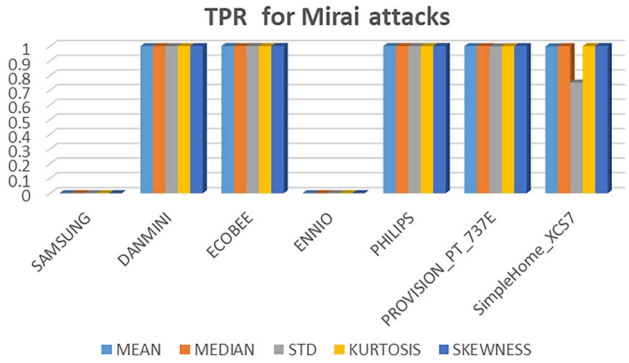

Fig. 31 TPR comparisons graph of Mirai attack

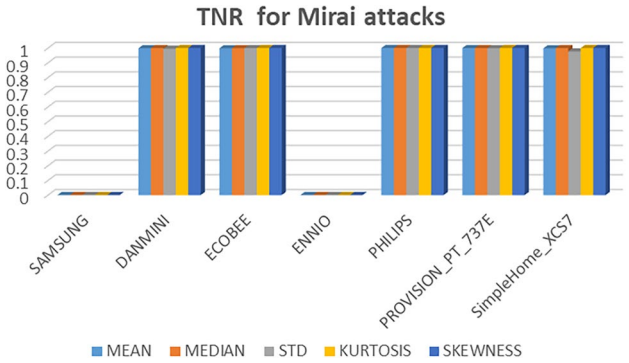

Fig. 32 TNR comparisons graph of Mirai attack

$$(\hat{Y}_j^{dm})_{h,i} = (E_l^{(dm)})_{h,i} + \sum_{k=1}^{q_1^{(dm-1)}} (q_{j,k}^{(dm)} * \hat{Y}_k^{(dm-1)})_{h,i} \quad (10)$$

$$= (E_j^{(dm)})_{h,i} + \sum_{k=1}^{q_1^{(dm-1)}} \sum_{e=-t_1^{dm}}^{t_1^{dm}} \sum_{f=-t_2^{dm}}^{t_2^{dm}} (Q_{j,k}^{(dm)})_{e,f} (\hat{Y}_k^{(dm-1)})_{h+e,i+f} \quad (11)$$

In this case, $Q_{j,k}^{(dm)}$ is the connection's adaptable load, and E_j^{dm} is the biased 2D array. Subsampling is used to assess the v_1^{dm} and v_2^{dm} skipping coefficients. Before applying the filter, the basic concept is to set the pixel count in both the longitudinal and transverse directions. While utilizing the skip rate, Eq. 12 is utilized to compute the dimension of the output feature maps.

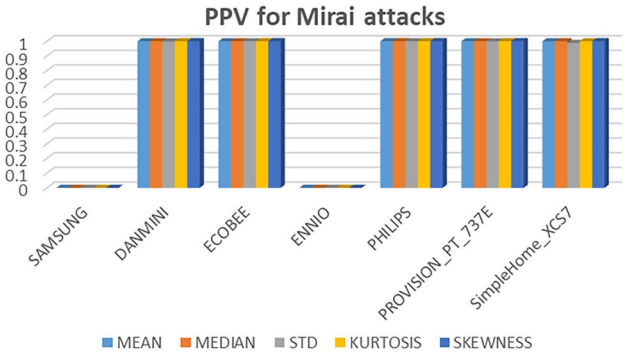

Fig. 33 PPV comparisons graph of Mirai attack

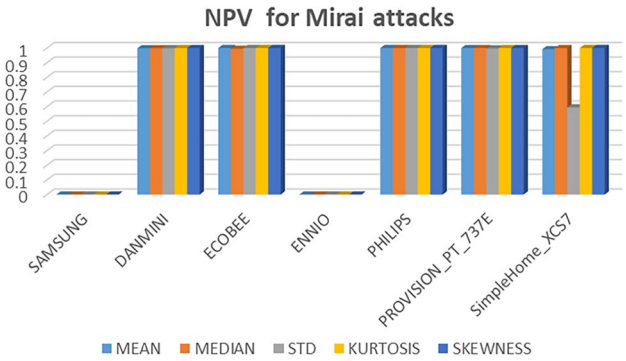

Fig. 34 NPV comparisons graph of Mirai attack

$$q_2^{dm} = \frac{q_2^{(dm-1)} - 2t_1^{dm})}{v_1^{dm} + 1} \quad and \quad q_3^{dm} = \frac{q_3^{(dm-1)} - 2t_2^{dm}}{v_2^{dm} + 1} \tag{12}$$

If dm be a non-linearity layer, with $\hat{Y}_j^{dm}$ 1 feature maps as input and $q_1^{dm} = q_1^{(dm-1)}$ feature maps as output, with $q_2^{(dm-1)} \times q_3^{(dm-1)}$ as the dimension of each, as stated in Eq. 13.

$$\hat{Y}_j^{dm} = g\left(\hat{Y}_j^{(dm-1)}\right) \tag{13}$$

The activation function in layer dm is denoted by the letter g, and it operates on a point-by-point basis. The suggested modified optimization method is used in this paper to efficiently tuning the activation function. Equation 14 is used to calculate the additional gain coefficient.

$$\hat{Y}_j^{dm} = hb_j g\left(\hat{Y}_j^{(dm-1)}\right) \tag{14}$$

Consider the correction layer to be dm. With the feature maps, each element does have an exact value and therefore is assessed using Eq. 15 with $q_1^{(dm-1)}$ feature map each of size $q_2^{(dm-1)} \times q_3^{(dm-1)}$ as an input.

$$\hat{Y}_j^{dm} = \left|\hat{Y}_j^{dm}\right| \tag{15}$$

Table 13 FPR comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0	0.001720324	0.001881822	0	0	0.00026959	0.002143623
MEDIAN	0.00E+00	2.46E−04	1.13E−03	0.00E+00	0.00E+00	5.39E−04	3.57E−04
STD	0.00E+00	4.79E−03	7.53E−04	0.00E+00	3.04E−05	1.08E−03	2.35E−02
KURTOSIS	0	0	0	0	0	0	0.00011909
SKEWNESS	0	0	0	0	0	0	0

Table 14 FNR comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0	0.000736016	0	0	0	0.000166431	0.003161739
MEDIAN	0.00E+00	8.34E−04	6.19E−04	0.00E+00	0.00E+00	8.32E−05	2.67E−04
STD	0	0.00063788	0	0	0	0.002995756	0.248619523
KURTOSIS	0	9.81354E−05	0	0	0	0.000166431	0
SKEWNESS	0	9.81354E−05	0	0	0	8.32154E−05	0
MEAN	0	0.000736016	0	0	0	0.000166431	0.003161739
MEDIAN	0.00E+00	8.34E−04	6.19E−04	0.00E+00	0.00E+00	8.32E−05	2.67E−04
STD	0	0.00063788	0	0	0	0.002995756	0.248619523
KURTOSIS	0	9.81354E−05	0	0	0	0.000166431	0
SKEWNESS	0	9.81354E−05	0	0	0	8.32154E−05	0

Table 15 FDR comparisons of Mirai attack

	App1	App2	App3	App4	App5	App6	App7
MEAN	0.999833569	0.000686982	0.000220965	0	0	0.000249626	0.000803464
MEDIAN	0.00E+00	9.82E−05	1.33E−04	0.00E+00	0.00E+00	4.99E−04	1.34E−04
STD	0	0.001911203	8.83978E−05	0	5.43419E−05	0.001000584	0.011540715
KURTOSIS	0	0	0	0	0	0	4.45295E−05
SKEWNESS	0	0	0	0	0	0	0

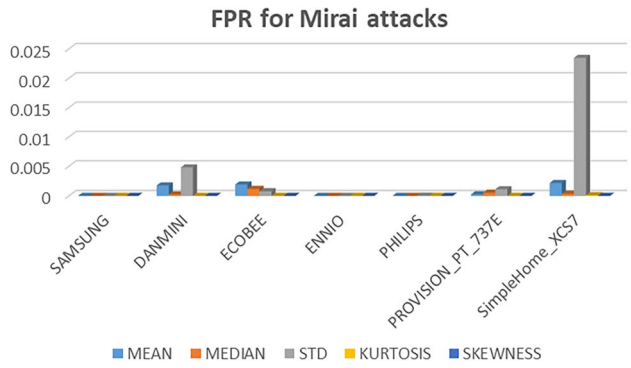

Fig. 35 FPR comparisons graph of Mirai attack

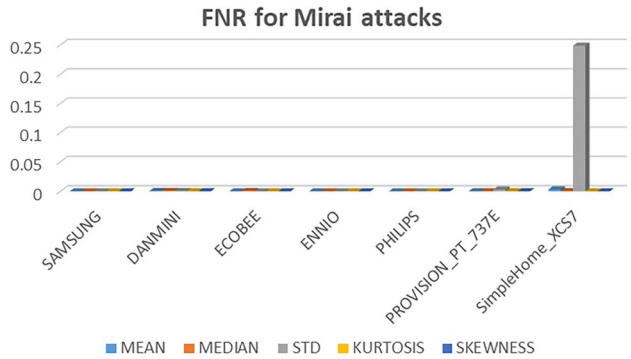

Fig. 36 FNR comparisons graph of Mirai attack

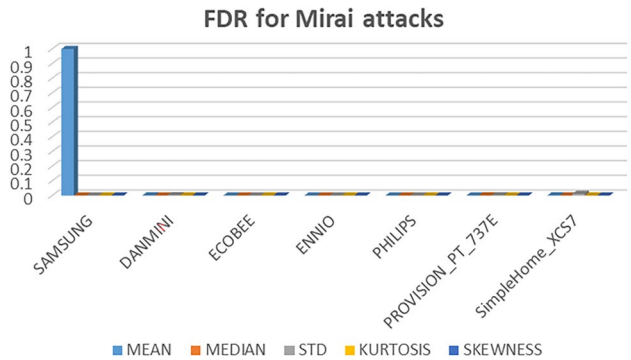

Fig. 37 FDR comparisons graph of Mirai attack

The output has the $q_1^{dm} = q_1^{(dm-1)}$ feature maps without any change in size because the absolute value is assessed an order to enhance.

Using *dm* as the pooling layer, and results consisting of $q_1^{dm} = q_1^{(dm-1)}$ feature maps with the smallest size. Pooling allows for the subsampling of feature maps by positioning the viewing windows at distinct places on every characteristic map and keeping a single value for each window. This layer distinguishes between two types of pooling as following.

When the boxcar filter is used, the procedure is known as Average Pooling and is denoted by the letters $R_{average}$

Every window's maximum value is considered to still be in max-pooling and is represented utilizing $R_{maximum}$

Suppose that *dm* is the convolutional layer. If the level $dm-1$ is not properly configured, the layer *dm* receives input apart from $q_1^{(dm-1)}$ feature maps with sizes of $q_2^{(dm-1)} \times q_3^{(dm-1)}$, and the k level having j^{th}-th unit is assessed according to Eq. 16.

$$\begin{aligned} \hat{y}_j^{dm} &= g\left(w_j^{dm}\right) \; with \; w_j^{dm} \\ &= \sum_{k=1}^{q_1^{dm-1}} \sum_{h=1}^{q_2^{dm-1}} \sum_{i=1}^{q_3^{dm-1}} X_{j,k,h,i}^{dm} \left(\hat{Y}_k^{(dm-1)}\right) \end{aligned} \tag{16}$$

4 Optimized performance to Resolve Difficulties with Optimization

4.1 The Solution Encode

The paper offers a new revolutionary updated technique that fine-tunes specific Convolutional Neural Network parameters in order to achieve accurate identification of an attack. Here, Q_δ denotes the number of filtering in the convolution level, Q_T is the size of the filter, and g is the transfer function. Q_δ and Q_T are almost certainly in the 1 to 25 range. This activation function varies depending on the performance of each of the nine apps employed in this study.

$$objective = min(error) \tag{17}$$

Table 16 Normalization and feature selection-guided TPR comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.881896809	0.90349479	0.912037037	0.002119243	0.98829111	0.01954955	0.91147045
Mutual Info classif	0.998608212	0.99938688	0.999614198	1	0.999908762	0.998918919	0.999767334
N and mutual Info	1	1	1	0.99901102	0.999817524	1	1

Table 17 Normalization and feature selection-guided TNR Comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.999223267	0.99991574	0.999717859	0.999622891	0.255815928	0.9995905	0.999445369
Mutual Info classif	0.998964357	0.99941018	0.998965485	0.998397285	0.998301919	0.998935299	0.998890738
N and mutual info	0.999309571	0.99974722	0.999905953	0.999622891	0.999660384	0.998361998	0.99935293

Table 18 Normalization and feature selection-guided PPV comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.998986486	0.9998643	0.998732573	0.789473684	0.787571799	0.977477477	0.99923479
Mutual Info classif	0.998806801	0.99914184	0.995772483	0.997603946	0.99939206	0.998828934	0.998605624
N and mutual info	0.999205324	0.99963226	0.999614346	0.999434629	0.999878342	0.998201439	0.99918633

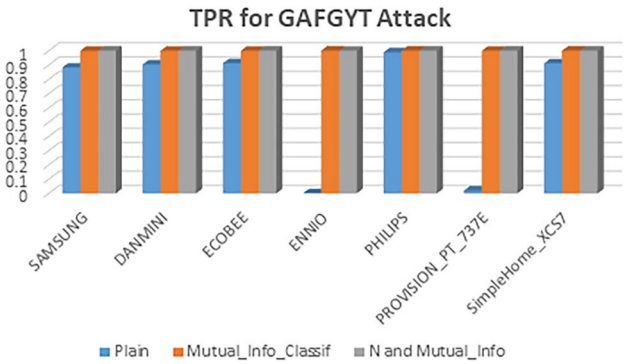

Fig. 38 Normalization and feature selection-guided TPR comparisons graph of GAFGYT attack

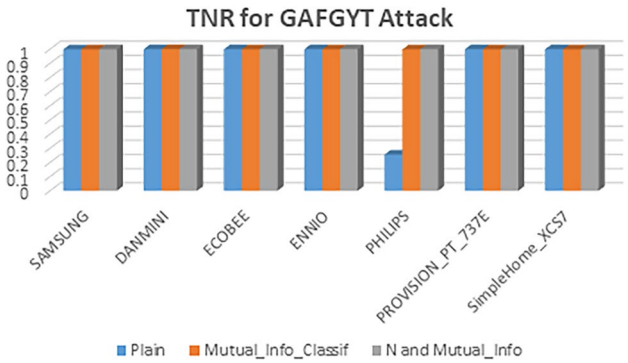

Fig. 39 Normalization and feature selection-guided TNR comparisons graph of GAFGYT attack

4.2 Method for Improved Optimization

Imagine a projectile going in a homogeneity, directed gravity field, with a non-zero beginning velocity in the horizontal direction, according to Newton's law. The projectile that was ejected at an edge Ω has supplied the velocity g (the stagnation point and the gravity gradient direction are perpendicular to one another) and is starting to move in the parabolic direction as shown in Eq. 18, within the coordinate (m, c), where the acceleration due to gravity is embodied as hs.

$$c = uh\Omega.m - \frac{hs.m^2}{2.w_0^2.\cos^2\Omega} \tag{18}$$

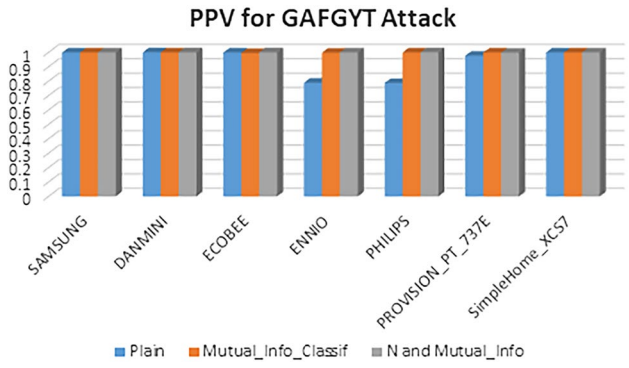

Fig. 40 Normalization and feature selection-guided PPV comparisons graph of GAFGYT attack

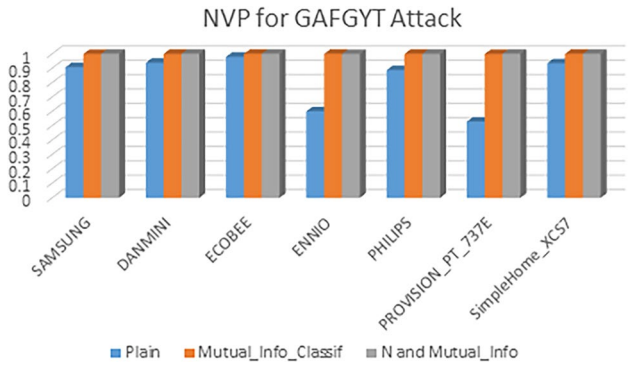

Fig. 41 Normalization and feature selection-guided NVP comparisons graph of GAFGYT attack

For just a clear answer, the suggested modified optimization algorithm has the following steps:

1. Pick an angle value of Ω_0 at randomly.
2. Adjust the count of iterations to $j = 0$ and substitute $\Omega^j = \Omega_0$for the goal function value $G_{object}(\Omega_0)$
3. Draw a correction angle: $\lambda^j \lambda^j > 0,\ hs(\lambda^j) = \left(\cos\left(\lambda^j\right)\right)^{-1}$ for $\lambda^j \leq 0, hs\left(\lambda^j\right) = \cos\left(\lambda^j\right)$.
4. Sketch a correction angle ρ^j, for $\rho^j > 0,\ hs\left(\rho^j\right) = \left(\cos\left(\rho^j\right)\right)^{-1}$, for $\rho^j \leq 0,\ hs\left(\rho^j\right) = \cos\left(\rho^j\right)$
5. Calculate the adjusted angle of the solution. A new plan is used in this step: firstly, a random vector s is given, as well as a threshold lets say, 0.5. If s increases a certain given threshold value, the adjusted angles is estimated using Eq. 19. In all other cases, the estimate of the adjusted angle is relied on Eq. 20.

$$\Omega^{j+1} = \Omega^j . hs\left(\lambda^j\right) . hs\left(\rho^j\right) \tag{19}$$

$$\Omega^{j+1} = y_{best} + \left(s\ x\ \Omega^j\right) \tag{20}$$

6. Asses the objective function value $G_{object}\left(\Omega^{j+1}\right)$.
7. The calculation is completed if $|G_{object}\left(\Omega^{j+1}\right) - G_{object}\left(\Omega_0\right)| < \zeta$ Alternatively, proceed to step 3, where the condition utilized to finish the computation is known as ζ.
8. $\Omega_t = \Omega^{j+1}$ is the optimum angle.

Table 19 Normalization and feature selection-guided NPV comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.90694031	0.93780623	0.979001658	0.60019246	0.886698058	0.528629591	0.934243498
Mutual info classif	0.998791958	0.99957863	0.999905865	1	0.99974492	0.999017119	0.999814952
N and mutual info	1	1	1	0.999340245	0.999490662	1	1

Table 20 Normalization and feature selection-guided FPR comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.000776733	8.426019548e	0.000282141	0.00037	0.744184072	0.0004095	0.00055463
Mutual info classif	0.001035643	0.00058982	0.001034515	0.001602715	0.001698081	0.001064701	0.001109262
N and mutual Info	0.000690429	0.00025278	9.40E−05	0.000377109	0.000339616	0.001638002	0.00064707

Table 21 Normalization and feature selection-guided FNR comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.118103191	0.09650521	0.087962963	0.997880757	0.01170889	0.98045045	0.088529549
Mutual info classif	0.001391788	0.00061312	0.000385802	0	9.12E−05	0.001081081	0.000232666
N and mutual info	0	0	0	0.00098898	0.000182476	0	0

Table 22 Normalization and feature selection-guided FDR comparisons of GAFGYT attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0.001013514	0.0001357	0.001267427	0.210526316	0.212428201	0.022522523	0.000765209
Mutual info classif	0.001193199	0.00085816	0.004227517	0.002396054	0.00060794	0.001171066	0.001394376
N and mutual info	0.000794676	0.00036774	0.000385654	0.000565371	0.000121658	0.001798561	0.00081367

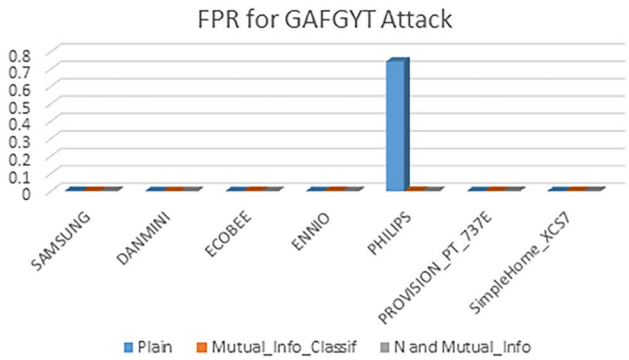

Fig. 42 Normalization and feature selection-guided FPR comparisons graph of GAFGYT attack

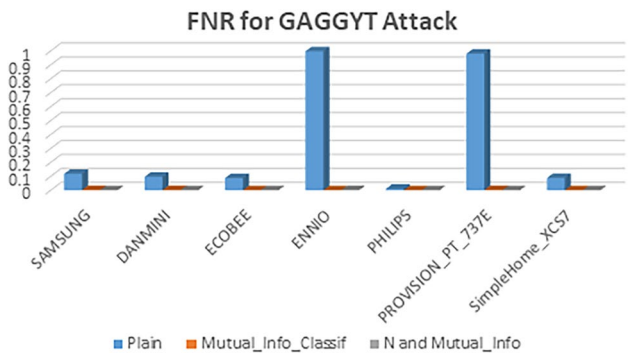

Fig. 43 Normalization and feature selection-guided FNR comparisons graph of GAFGYT attack

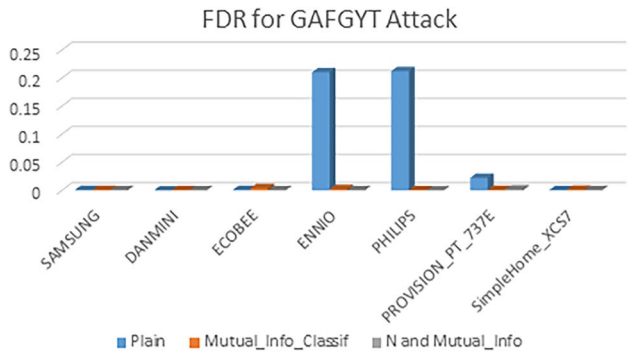

Fig. 44 Normalization and feature selection-guided FDR comparisons graph of GAFGYT attack

5 Results and analysis

Python was used to implement the proposed attack detection system. archiveicsuciedu (2021) was used to download the seven programmes used in this study. Under Mirai and GAF-GYT attacks, two different calculations were done. The efficiency of the proposed approach was also compared to those of other existing approaches like FAE-GWO-DBN, AIG (Pijarski and Kacejko 2019), and DCNN (Li et al. 2020). Furthermore, the discussion included both positive and negative measures. Reliability, sensitivities, clarity, and specificity, as well as NPV, MCC, and F_1-score, are positive measurements, while FPR, FNR, and FDR are negative measures.

Table 23 Normalization and feature selection-guided TPR comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	0.005406734	0.846066993	0	0.990320224	0.998831776	0.914374181
Mutual info classif	0	0.005406734	0.846066993	0	0.990320224	0.998831776	0.914374181
N and mutual Info	0	1	1	0	1	0.99946082	1

Table 24 Normalization and feature selection-guided TNR comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	0.999901865	1	0	0.081354274	0.446367646	0.99942109
Mutual info classif	0	1	1	0	0.99989131	0.999833569	0.999510153
N and mutual info	0	1	1	0	1	0.999916785	0.999955468

Table 25 Normalization and feature selection-guided PPV comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	0.956521739	1	0	0.658076783	0.62556281	0.998309713
Mutual info classif	0	1	1	0	0.999938584	0.999819641	0.998682792
N and mutual info	0	1	1	0	1	0.999910096	0.999880924

Table 26 Normalization and feature selection-guided NPV comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	0.715720708	0.982242098	0	0.824793388	0.997582295	0.96895778
Mutual info classif	0	0.998872715	0.998411227	0	0.984535531	0.996599204	0.997466892
N and mutual info	0	1	1	0	1	0.999500915	1

Table 27 Normalization and feature selection-guided FPR comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	9.813542688e	0	0	0.918645726	0.553632354	0.00057891
Mutual info classif	0	0	0	0	0.00010869	0.057085795	0.00017813
N and mutual info	0	0	0	0	0.00010869	0.057085795	0.00017813

Table 28 Normalization and feature selection-guided FNR comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	0.994593266	0.153933007	0	0.009679776	0.001168224	0.08562582
Mutual info classif	0	0.002826247	0.013549116	0	0.008797029	0.0036844	0.006788139
N and mutual info	0	0	0	0	0	0.00053918	0

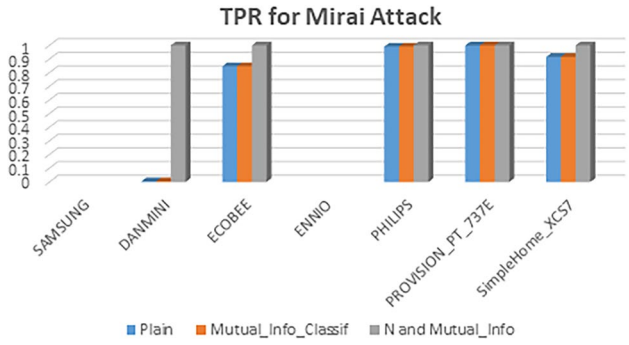

Fig. 45 Normalization and feature selection-guided TPR comparisons graph of Mirai attack

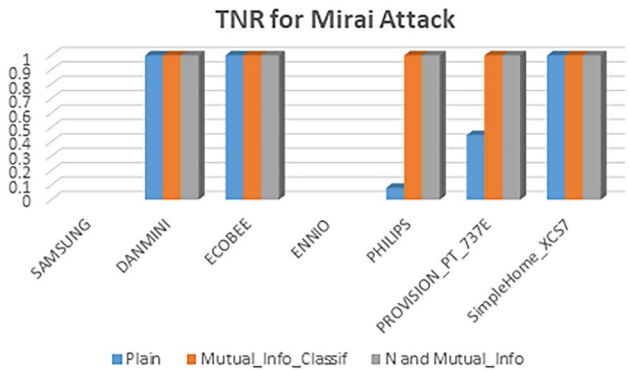

Fig. 46 Normalization and feature selection-guided TNR comparisons graph of Mirai attack

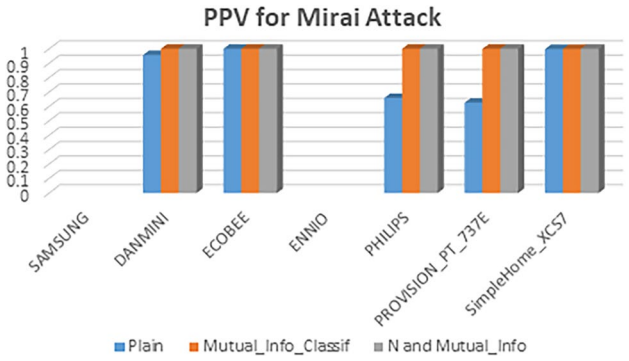

Fig. 47 Normalization and feature selection-guided PPV comparisons graph of Mirai attack

The proposed method is assessed using positive performance indicators under the observation of the Mirai assault for seven applications (Figs. 4, 5, 6, 7, 8, 9, 10). In actuality, the performance is said to be greater if they retain optimum value in comparison to other existing models. The proposed method for application 3 obtains improved accuracy, with 60.14%, 3.10%, and 5.46% higher consistency than DCNN, Algorithm of the Innovative Gunner and FAE-GWO-DBN, correspondingly. Similarly, in terms of accuracy measurement, the proposed technique outperforms traditional models such as DCNN, Algorithm of the Innovative Gunner and FAE-GWO-DBN by 69.76%, 3.27%, and 22.68% accordingly. The recognized model's sensitivity spans between 98% to 99.9%, whereas other existing methods have a smaller containing compounds. Furthermore, for application 2, the established model is 11.06%, 3.72%, and 78.47% better than FAE-GWO-DBN, AIG, and DCNN accordingly, in terms of F_1-score in Figs. 8, 9 and 10. In terms of MCC, the suggested model outperforms previous comparable models, with a result of 98%-−99.9%. So far, the findings have been positive for other important outcomes and are examined for superior performance, validating the suggested work's improved performance.

The suggested model's performance is compared to that of standard models with respect to of negative metrics. Figures 11, 12 and 13 demonstrates that. The effectiveness of the constructed model is scrutinized in light of several negative measures during the Mirai attack. It is noticed that the smallest value of positive measurements demonstrates simple capital detection mechanism, that the suggested work satisfies. According to this, When the FPR is taken into account, the suggested layout for application 4 has the lowest FPR, which is 98.4%, 67.15%, and 91.46% higher than DCNN, Algorithm of the Innovative Gunner, and FAE-GWO-DBN, accordingly. In this proposed work, the FPR estimate under the Mirai assault has obtained the lowest magnitudes in terms of error, that are in the range of 0.00%–0.01%. The FNR and FDR error measures are also analyzed and examined for each of the nine instances. As a consequence, the desired outcomes are realized. As a consequence, the results show that previous work on these low error metrics has improved.

Figures 14, 15, 16, 17, 18, 19 and 20 show the performance of the community center during the identification

Table 29 Normalization and feature selection-guided FDR comparisons of Mirai attack

	SAMSUNG	DANMINI	ECOBEE	ENNIO	PHILIPS	PROVISION _PT_737E	SimpleHome _XCS7
Without normalizaion and feature selection	0	0.043478261	0	0	0.341923217	0.37443719	0.00169029
Mutual info classif	0	0	0	0	6.14E−05	0.000180359	0.001317208
N and mutual info	0	0	0	0	0	8.99E−05	0.000119076

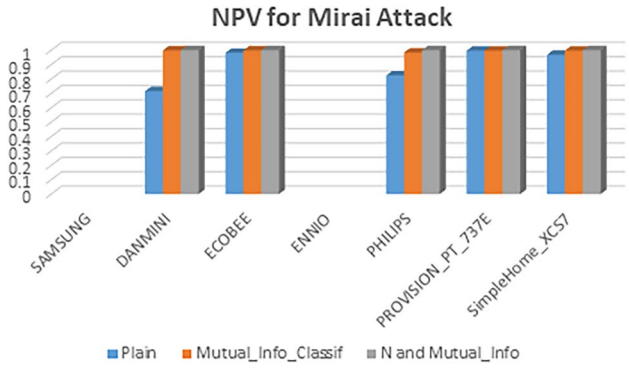

Fig. 48 Normalization and feature selection-guided NPV comparisons graph of Mirai attack

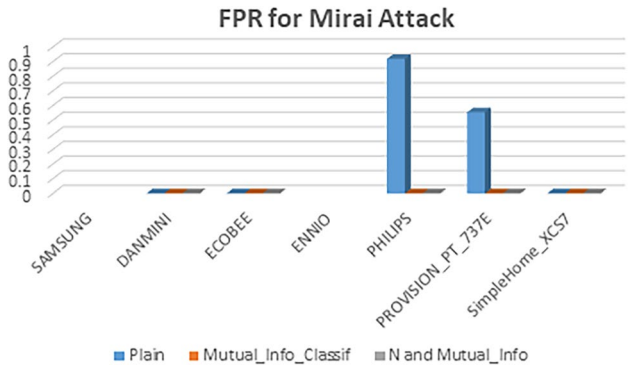

Fig. 49 Normalization and feature selection-guided FPR comparisons graph of Mirai attack

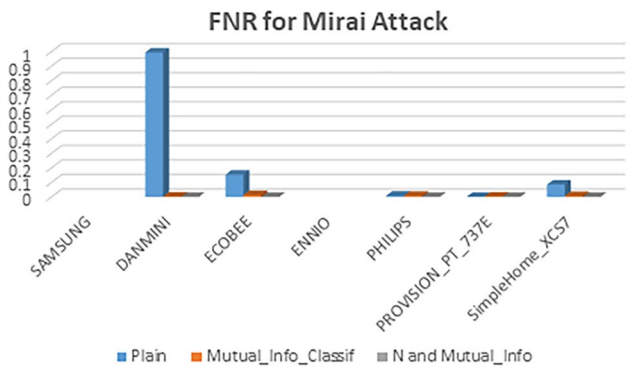

Fig. 50 Normalization and feature selection-guided FNR comparisons graph of Mirai attack

of the GAF-GYT assault. For each of the seven applications, the suggested work is evaluated against standard terms. The greatest value of a positive measure automatically indicates that the system's situation has improved. Under these settings, the established model for classification accuracy is 65.34%, 3.02%, and 4.14% better than the

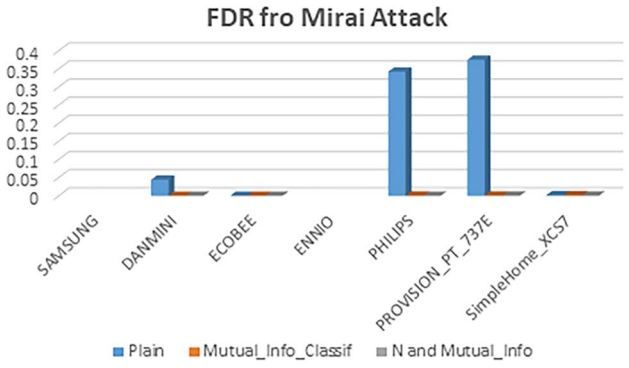

Fig. 51 Normalization and feature selection-guided FDR comparisons graph of Mirai attack

standard model for application 1 from DCNN, Algorithm of the Innovative Gunner and FAE-GWO-DBN, accordingly. However, using the sensitive measurement, the suggested work's effectiveness in applications 5 is 24.09%, 4.98%, and 22.34% better than DCNN, Algorithm of the Innovative Gunner, and FAEGWO-DBN, accordingly. In terms of precision, the created model achieves higher average value than other standard terms, ranging from 97% to 99.99%. For all the other applications requiring positive measures, the entire performance is evaluated, and the resulting charts are produced. Overall, the findings show that the suggested approach outperforms other conventional approaches on all good criteria.

Figures 21, 22 and 23 depicts the suggested work's effectiveness against other comparable method in terms of some unfavorable measures. For all nine applications using the negative measure, the existing study is assessed under the identification of the GAF-GYT assault. For the FDR measure of application 3, the proposed model outperforms the comparison methods with the lowest FDR value, which are 87.91%, 57.40%, and 88.67% greater to DCNN, Algorithm of the Innovative Gunner, and FAE-GWO-DBN, respectively. Furthermore, the FPR magnitude of the suggested approach is modest, averaging around 0.01%, whereas other existing methods perform poorly with higher FPR values. Overall, the suggested method outperforms the competition in terms of preventing malicious involving negative indicators.

Tables 2, 3, 4, 5, 6, 7 and 8 and Fig. 24, 25, 26, 27 , 28, 29 and 30 shows the comparisons of the performances of mean, median, standard deviation, Kurtosis and Skewness when used as feature selection for calculating TPR, TNR, PPV, NPV, FPR, FNR, and FDR respectively for GAFGYT attack.

Tables 9, 10, 11, 12, 13, 14 and 15 and Figs. 31, 32, 33, 34, 35, 36 and 37 shows the comparisons of the performances of mean, median, standard deviation, Kurtosis

and Skewness when used as feature selection for calculating TPR, TNR, PPV, NPV, FPR, FNR, and FDR respectively for Mirai attack.

Tables 16, 17, 18, 19, 20, 21 and 22 and Figs. 38, 39, 40, 41, 42, 43 and 44 shows the comparisons of the performances of without normalization and feature selection, Mutual_Info_Classif (Only Mutual_Info_classif feature selection but no normalization method), N and Mutual_Info_classif (Info_classif feature selection with Normalization) for calculating TPR, TNR, PPV, NPV, FPR, FNR, and FDR respectively for GAFGYT attack.

Tables 23, 24, 25, 26, 27, 28 and 29 and Figs. 45, 46, 47, 48, 49, 50 and 51 shows the comparisons of the performances of without normalization and feature selection, Mutual_info_classif (Only Mutual_info_classif feature selection but no Normalization method), N and Mutual_Info (F_classif feature selection with Normalization) for calculating TPR, TNR, PPV, NPV, FPR, FNR, and FDR respectively for Mirai attack.

6 Conclusions and future scope

A unique intrusion detection method was introduced in this work by interlinking the DevOps architecture with 2 steps: extraction of features and classifying of them. The data processing from each application was done in the early stages of feature extraction by combining the statistics and higher-order descriptors with the existing features. Moreover, an optimized DCNN approach was used to develop the classification process using these retrieved features. Furthermore, a unique method was used to optimize the number of filtering and size of the filters in the fully connected layers, also the input vector. This study describes a method for detecting attacks on WSN. A unique method is used to deal with the optimization concerns. Furthermore, the adopted work's performance is much better in comparison to that of other traditional models in terms of accuracy, FNR, sensitivity, MCC, specificity, FDR, FPR, and NPV, F_1-score under the GAF-GYT as well as Mirai attacks. In terms of negative measurements, it can be demonstrated that the model developed performs more effectively when contrasting the suggested approach to the latest techniques for recognizing assaults. This is due to the suggested algorithm's quick pace in tackling diverse optimization problems. It also has a high level of quality. This paper compares the performances of without normalization and feature selection, F_Classif (Only F_classif feature selection but no Normalization method), N and F_classif (F_classif feature selection with Normalization) for calculating TPR, TNR, PPV, NPV, FPR, FNR, and FDR respectively for GAF-GYT and Mirai attacks. In the case of application 3, the developed approach outperforms the DCNN, Algorithm of the Innovative Gunner, and FAE-GWO-DBN methods by 60.14%, 3.10%, and 5.46%, accordingly. Furthermore, the suggested model for applications four achieves a low FPR, which is better than FAE-GWO-DBN, AIG, AND DCNN techniques by 91.46%, 67.15%, and 98.4%, respectively. Furthermore, the proposed technique outperforms the DCNN, Algorithm of the Innovative Gunner and FAE-GWO-DBN methods by 69.76%, 3.27%, and 22.68%, respectively. As a result, the improved results demonstrate the proposed algorithm's superiority to previous designs. Other deep learning approaches and metaheuristics algorithms may be applied in the future to increase the performance of intrusion detection systems.

Funding There is no funding involve in this research.

Declarations

Conflicts of interest No conflict of Interest.

References

Alauthman M, Aslam N, Al-kasassbeh M, Suleman Khan KK, Choo R (2020) An efficient reinforcement learning-based Botnet detection approach. J Netw Comput Appl 150(15):102479

Alfan G, Syafrudin M, Farooq U, Ma'arif MR, Rhee J (2020) Improving efficiency of RFID-based traceability system for perishable food by utilizing IoT sensors and machine learning model. Food Control 110:107016

archiveicsuciedu (2021) archive.ics.uci. https://archive.ics.uci.edu/ml/datasets/detection_of_IoT_botnet_attacks_N_BaIoT#

Asadi M, Ali M, Jamali J, Parsa S, Majidnezhad V (2020) Detecting botnet by using particle swarm optimization algorithm based on voting system. Futur Gener Comput Syst 107:95–111

Azar J, Makhoul A, Barhamgi M, Couturier R (2019) An energy efficient IoT data compression approach for edge machine learning. Futur Gener Comput Syst 96:168–175

Baig ZA, Sanguanpong S, Naeem Firdous S, Nhan Vo V, So-In C (2020) Averaged dependence estimators for DoS attack detection in IoT networks. Futur Gener Comput Syst 102:198–209

Chen Y, Kintis P, Antonakakis M, Nadji Y, Farrell M (2017) Measuring lower bounds of the financial abuse to online advertisers: a four year case study of the TDSS/TDL4 Botnet. Comput Secur 67:164–180

Cheng JCP, Chen W, Chen K, Wang Q (2020) Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms. Autom Construct 112:103087

Giridhar Reddy B, Sai Ambati L (2020) A novel framework for crop pests and disease identification using social media. In: MWAIS 2020 proceedings 9

Hasan M, Islam M, Zarif I, Hashem MMA (2019) Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. Internet Things 7:100059

Ho J (2018) Efficient and robust detection of code-reuse attacks through probabilistic packet inspection in industrial IoT device. IEEE Access 6:54343–54354

Jung W, Zhao H, Sun M, Zhou G (2020) IoT botnet detection via power consumption modelling. Smart Health 15:100–103

Klassen M, Yang N (2012) Anomaly based intrusion detection in wireless networks using Bayesian classifier. In: 2012 IEEE fifth international conference on advanced computational intelligence (ICACI)

Koroniotis N, Moustafa N, Sitnikova E, Turnbull B (2019) Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. Futur Gener Comput Syst 100:779–796

Li Y, Yingying X, Liu Z, Hou H, Cui L (2020) Robust detection for network intrusion of industrial IoT based on multi-CNN fusion. Measurement 154(15):107450

Liu L, Ma Z, Meng W (1989) Detection of multiple-mix-attack malicious nodes using perceptron-based trust in IoT networks. Futur Gener Comput Syst 101:865–879

Maleh Y, Ezzati A, Qasmaoui Y, Mbida M (2015) A global hybrid intrusion detection system for wireless sensor networks. Procedia Comput Sci 52:1047–1052. https://doi.org/10.1016/j.procs.2015.05.108

Mourabit YE, Toumanari A, Bouirden A, Moussaid NE (2015) Intrusion detection techniques in wireless sensor network using data mining algorithms: comparative evaluation based on attacks detection. Int J Adv Comput Sci Appl. https://doi.org/10.14569/IJACSA.2015.060922

Mousavi SH, Khansari M, Rahmani R (2020) A fully scalable big data framework for Botnet detection based on network trafic analysis. Inf Sci 512:629–640

Murali S, Jamalipour A (2020) A lightweight intrusion detection for Sybil attack under mobile RPL in the Internet of Things. IEEE Internet Things J 7(1):379–388

Nguyen HT, Ngo QD, Nguyen DH, Le Van-Hoang (2020) PSI-rooted subgraph: a novel feature for IoT botnet detection using classifier algorithms. ICT Express 6(2):128–138. https://doi.org/10.1016/j.icte.2019.12.001

Pijarski P, Kacejko P (2019) A new metaheuristic optimization method: the algorithm of the innovative gunner (AIG). Eng Optim 51(12):2049–2068

Pour MS, Mangino A, Friday K, Rathbun M, Ghan N (2020) On data-driven curation, learning and analysis for inferring evolving internet-of-things (IoT) botnets in the wild. Comput Secur 91:101707

Sandhya G, Julian A (2014) Intrusion detection in wireless sensor networks using genetic K-means algorithm. In: 2014 IEEE international conference on advanced communications, control and computing technologies

Sarma SK (2021) Optimally configured deep convolutional neural network for attack detection in internet of things: impact of algorithm of the innovative gunner. Wireless Pers Commun 118:239–260

Shafq M, Tian Z, Sun Y, Xiaojiang D (2020) Selection of effective machine learning algorithm and Bot-IoT attacks traffic identification for internet of things in smart city. Futur Gener Comput Syst 107:433–442

Shailendra Rathore J, Park H (2018) Semi-supervised learning based distributed attack detection framework for IoT. Appl Soft Comput 72:79–89

Shaon MNA, Ferens K (2015) Wireless sensor network wormhole detection using an artificial neural network. In: International conference of wireless networks. Las Vegas, USA, pp 115–120

Sharma S, Singh H, Sarkar M, A (2023) Detection of Mirai and GAF-GYT attack in wireless sensor network. In: Hemanth, J, Pelusi, D, Chen, IZ J (eds) Cyber physical systems and internet of things. ICoICI 2022. Engineering cyber-physical systems and critical infrastructures, Springer, vol 3

Sherazi HHR, Iqbal R, Ahmad F, Khan ZA, Chaudary MH (2019) DDoS attack detection: a key enabler for sustainable communication in internet of vehicles. Sustain Comput Inf Syst 23:13–20. https://doi.org/10.1016/j.suscom.2019.05.002

Singh M, Dutta N, Singh TR, Nandi U (2020) A technique to detect wormhole attack in wireless sensor network using artificial neural network. In: Suma V, et al (eds) Evolutionary computing and mobile sustainable networks, Lecture notes on data engineering and communications technologies, Springer, Singapore, vol 53, pp 297–307, https://doi.org/10.1007/978-981-15-5258-8_29

Singh R, Singh J (2017) Fuzzy based advanced hybrid intrusion detection system to detect malicious nodes in wireless sensor networks. Wirel Commun Mobile Comput. https://doi.org/10.1155/2017/3548607