Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.034885
Article

Tech Science Press

A Cyber-Attack Detection System Using Late Fusion Aggregation Enabled Cyber-Net

P. Shanmuga Prabha* and S. Magesh Kumar

Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (Deemed to be University), Chennai, Tamilnadu, India
*Corresponding Author: P. Shanmuga Prabha. Email: shanmugapabhap.sse@saveetha.com

Received: 30 July 2022; Accepted: 15 December 2022

Abstract: Today, securing devices connected to the internet is challenging as security threats are generated through various sources. The protection of cyber-physical systems from external attacks is a primary task. The presented method is planned on the prime motive of detecting cybersecurity attacks and their impacted parameters. The proposed architecture employs the LYSIS dataset and formulates Multi Variant Exploratory Data Analysis (MEDA) through Principle Component Analysis (PCA) and Singular Value Decomposition (SVD) for the extraction of unique parameters. The feature mappings are analyzed with Recurrent 2 Convolutional Neural Network (R2CNN) and Gradient Boost Regression (GBR) to identify the maximum correlation. Novel Late Fusion Aggregation enabled with Cyber-Net (LFAEC) is the robust derived algorithm. The quantitative analysis uses predicted threat points with actual threat variables from which mean and difference vectors are evaluated. The performance of the presented system is assessed against the parameters such as Accuracy, Precision, Recall, and F1 Score. The proposed method outperformed by 98% to 100% in all quality measures compared to existing methods.

Keywords: External attacks; cyber-physical systems; principle component analysis; singular value decomposition; recurrent 2 convolutional neural network; gradient boost regression

1 Introduction

The cyber-physical systems control the mechanism monitored by in-built pre-programmed algorithms with the help of the internet. Cyber-physical systems are integrated with robust, authenticated environments to achieve high safety mechanisms. Autonomous systems, smart grids, and automobile systems are typical cyber-physical systems. The insertion of external attacks is a common occurrence of insecure issues. It is essential to evaluate user-related data engendered in the description model, such as user access duration and quality report. These data records reflect the user gauges regarding user integrity, inspiration, and specialized quality [1].

IoT-enabled networks create these cyber-physical systems vulnerable to malicious attacks such as Denial-of-Service (DoS) attacks and deception attacks. These malicious attackers not only attack internet-activated network nodes but also directly affect the physical systems [2]. Therefore, it is essential to model an efficient control approach for assuring cyber-physical system security in the presence of numerous malicious attacks. The most common attack is the DoS attack which does not require any system information and will root the most economically expensive security occurrences. For instance, when a critical industrial control process becomes open-loop unstable, the DoS attack can result in environmental damage. The most commonly occurring cyber threats are Social Engineered Trojans, Unpatched Software, Phishing, Network worms, etc. The Cyberattack types such as BOTS are the more significant part of low-secure conceived IoT gadgets, which are touchy to Botnets [3].

Cybersecurity system needs fundamental principles to hold a set of rules for accessing and restrictions on a specific type of information entry. The user information stored in the cyber protective system is encapsulated with an authorized shield process [4]. The integrity of the cloud platform is determined by assuring the trustworthiness of data and the accuracy of given information. Far away recording is the most difficult heaps of classified business conversations. Availability is another factor that cyber-physical systems access reliable information globally with immediate response. Regardless of device, network, and other factors, the network devices are connected via common communication nodes. The intrusion can come from any location. Security is demandable at communication devices and gateways [5].

Man-in-Middle Attacks: IoT gigantic gadgets are ongoing frameworks that generally happen in modern PCs, savvy gear-independent vehicles, etc. Denial of Services: The popular IoT gadgets are more inclined to hacking exercises, and DoS assaults center around IoT gadgets appropriately [6]. The present research is focused on understanding deep patterns of attacks occurring with cyber-physical systems during dynamic operations.

- The increase in cyber-physical systems, cloud migrations of IT infrastructure, and small enterprises rely on data storage, data transfer, and analysis. The digitally connected environment, with massive devices, increases the global supply chain attains complexity. The number of cyber-attacks will keep rising because of digitally acceptable activities.
- The research is initiated to learn more about the cyber-attack scenarios, the attack patterns, and the impacted parameters that directly hit the system. To reduce cyber-attacks, the pattern of distribution needs to be understood well. The cyber-attacks are external forces that initiate malicious activity by learning the present process and allowing the system to accept the malicious data without concerning the originality.
- This is a slow process, where cyber-attacks create multiple routes for entering the system. The right time is the lack of protection provided by the firewalls. The attack process overrides the virus shield and defender patterns to hit the system directly.
- The present research is focused on understanding the deep pattern of attack occurrence with cyber-physical systems and the cloud during dynamic operations. A novel algorithm is formulated based on the existing study made in Section 2. The drawbacks in the existing cybersecurity systems are propagation delay, attack patterns, and false statements are considered in the proposed research.
- The key takes away of the proposed work is to detect the cyberattacks as fast as possible and display the impacted parameter's reason for the attacks.

The novel algorithm is formulated based on the existing study explained in Section 2. The drawbacks in existing cybersecurity systems on propagation delay, attack patterns, and false statements are considered in the proposed research. The key objective of the proposed work is to detect the cyber-attacks as fast as possible and display the impacted parameter's reason for the attacks. The rest of the paper is organized as follows. Section 2 discusses the performance of related articles to cyberattack detection, and Section 3 describes the proposed framework in detail. Section 4 describes the experimental setup required for demonstrating the cyberattack detection system using the proposed framework. Section 5 discusses the results obtained with the presented system, and finally, Section 6 concludes this paper with necessary results and discussions related to challenges in the implementation process, etc.

2 Related Works

Sodagudi et al. proposed a Structured Query Language (SQL) injection in applications providing web services, which is a commonly occurring cyber-attack. The way that cyber-attacks target public computers is presented through consistent monitoring. Specifically, the attacks against the information on websites are simulated. Attacks such as SQL injection attacks, DNS attacks, and DoS attacks are considered for implementation [7]. Mestha et al. mentioned that cyber-attacks could occur in any operating system in the cyber-physical network. To determine a novel detection technique using a machine learning algorithm to identify malware is derived here. The resilient estimation of traditional learning objectives is presented with more precise steps. The presented estimators help identify the SQL injection problems [8]. Saharkhizan et al. presented an ensemble version of a cyber-attack detection system in IoT networks. An effective decision can be made using this module with a decision tree algorithm. The proposed approach uses a transparent deep learning model for detecting cyber-attacks in IoT systems. The present system achieves an accuracy of 98% with 65% of training [9].

The extreme gradient boosting algorithm detects malicious information attacks in self-driving vehicles. In the case of false data injection, the self-driven vehicles are more influenced by abnormal responses. XGB classifier is discussed in this presented system in which pattern classification accuracy of 92% is achieved [10]. The proposed model uses deep learning neural network to detect the pattern changes that occur slowly in the regular activities of the system. The resilient change is detected to identify the misbehaving applications inside the system. The author presented a feature selection for machine learning based on the early detection of distributed cyber-attacks. Principal component analysis and support vector machines compare performance with the random forest algorithm [11]. Machine learning algorithms are used to detect the malware present in the network effectively. A machine learning technique based on cyber threat and intrusion detection systems is enunciated [12]. Malware is a cyber threat that slows down the normal activity of the system and makes it vulnerable to applications present in cyber-physical systems [13]. Spam detection in cyber-physical systems is frequently detected and evaluated using spam classification [14,15]. Saxena et al. presented an approach for single and strong malicious attacks in the smart grid. The system monitors the malicious commands effectively. In a planned attack scenario, such as stealing devices, acquiring login credentials, and Distributed Denial of Services (DDoS) towards the communication node are highly proactive and cannot be detected by the existing system [16].

Barbeau et al. discussed a case of attack performed on sensors and actuators, which takes over the control of cyber-physical systems. The presented approach finds out the disruptions and bounces back from attacks. In this case, the hardware is forced to idle to safeguard the contents [17]. Sui et al. presented a system that detects vulnerable attacks in generating cyber-physical systems. The scenario makes bouncing changes in the detection of residuals after the attacks. The drawback is the hardware resilience because of the recurring changes [18]. Keshk et al. presented a comprehensive study on security in cyber-physical systems and discussed heterogeneous data sources handling problems in cyber-physical systems [19]. Heartfield et al. presented a cyber intrusion detection system in cyber-physical systems. The hyperparameters impact the sole discussion about various cyber-physical systems. The presented approach used Reinforced Learning techniques and could detect the issues of a smart home environment. Thus, the system is improved in accuracy, and various attacks evolve around the cyber-physical systems [20]. Di et al. presented a Deep belief network enabled by cyber-attack detection in industrial automation. The proposed framework detects malfunctions in the regular activities of the gas pipeline in an automated environment. The drawback that persists in the current scenario is the hidden parameters in the analysis unit not providing sufficient knowledge on the detection pattern of attacks [21]. Mousavinejad et al. discussed various impacts of cyber-physical in distributed systems. An attack detection mechanism is evaluated to implement the monitoring process in a timely manner and make adaptive changes to enhance the attacked system. The drawback is the lesser effectiveness of the proposed method [22]. Dong et al. presented a system that impacts traffic flow control systems by creating interruptions. The proposed model provides the idea of predicting cyber-attacks in traffic evaluations and analyzing the perspective of automated systems in network security. In this case, the drawback is the attacks' inefficient notability [23]. Joo et al. presented nonlinear signal integrity attacks over cyber-physical systems, resilience in malicious attacks, and robustness under the nonlinear environment. The drawback present in the system is the lack of process validation [24]. If the system is unaware of existing attack probes, the prolonged destruction of attacks corrupts the internal operations and provides a resilient change in the floor [25–27]. Bhatti et al. discussed various watermarking techniques in image processing, in which a novel structure using Quaternion Fourier Transform (QFT) is being implemented. The presented approach incorporates chaotic encryption, Arnold scrambling to achieve a high signal-to-noise ratio on images, and good Peak Signal to Noise (PSN) [28]. Various existing works discussed here focus on single parameter consideration, while the proposed work focuses on multiple parameters from the unbalanced data.

3 The Proposed Framework

3.1 System Design

Motivated by the above observations, this paper enunciates the proposed novel architecture using deep learning convolutional neural network with Multivariate Exploratory Data Analysis (MEDA) for effective processing. The data representation is modeled in the first stage using a multivariate analysis technique with Principal Component Analysis (PCA) and Singular Vector Decomposition (SVD). The late fusion technique is adapted, keeping multi-modality attributes in the given data. In the second stage cascaded recurrent convolutional neural network algorithm is evaluated with gradient boosters to detect the cyber-attacks. The initial raw data collected from the LYSIS data set is processed by evaluating the covariance matrix and given a vector formation. The multi-variant exploratory data analysis model (ensemble with PCA and SVD) elaborates the data for attributes highlighting the cyber-attack probes. The multi-modality issues occur during decision-making, which is considered an important scenario. Hence, the Late Fusion technique is employed for a recurrent weighted convolutional neural network, encapsulated with gradient boosters to generate app aggregated results. The performance of the entire model is evaluated through quantitative measures such as Accuracy, Precision, Recall, F1-Score, and Error rate.

3.2 LYSIS Dataset

LYSIS is the ultimate common platform for the social internet of things. The platform consists of cloud-based deployment models designed to provide the service. The social internet of things criteria are defined using social objects with visualization through various pre-programmed methodologies and reusable technology. The LYSIS data set consists of visual things, social things, and applications relevant to cloud environment services. The data set consists of various public devices, private devices, application models, and objects users connected with different locations in a certain area. The data set holds the device ID, name, and type of brand model. The object profile consists of application ID, application type, and device private information such as mobility ownership, device type, etc. These data are prepared before assigning to preprocessing. Using *the xlsread command, the dataset in XL file format is read* into the system model. Only integer values are extracted and assigned to separate the variables say d1, d2, d3 . . . d13, etc.

3.3 Implementation Summary

The summary of the proposed implementation is categorized into two phases. The first phase deals with multivariate data analysis in which the license data set is segregated with required attributes such as timestamp, source IP, destination IP, processing delay location, etc. The multivariate exploratory data analysis is the tangible structure of principal component analysis and singular vector decomposition, which requires identifying agent space vectors from the raw data. The unified practice is routed together to form the feature mappings as an array.

Further, the feature data is reshaped into a matrix with two-dimensional scales. This data is applied to recurrent convolutional neural network architecture tuned to extract unique patterns present in the given LYSIS dataset. The deep learning convolutional neural network architecture consists of an input layer of size 100 × 100 × 1 with a stride layer of 1 × 10. This is followed by a fully connected layer of size 350 × 1, and the repeated fully connected layer is channeled to correlate the input pattern with the trained pattern. The ReLu layer is used in the initial stage at the input layer and fully connected layer to obtain exponential growth protection.

The CNN architecture is trained and tested with several iterations. After ‘N’ epochs of the stochastic gradient training model, the optimized output is mapped using the accuracy plot. The recurrent convolutional neural network output provides the maximum correlation rate with different types of attacks and datasets. The present system architecture is developed using the LFAEC fusion technique in which deep convolutional neural network architecture and gradient regression model are fused together to make the final decision. The gradient-boosted regression model is developed, and the max score is evaluated from feature-extracted outputs. On the other hand, deep CNN is evaluated to produce the match scores. Fig. 1 represents the System architecture of the proposed LFAEC.

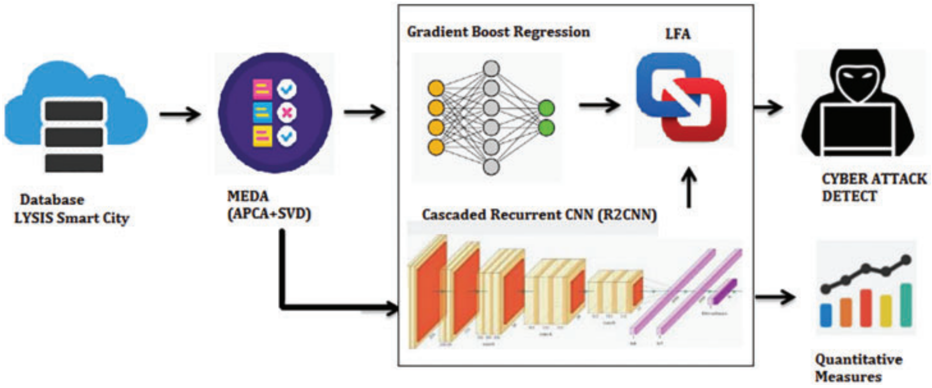


Figure 1: System architecture of proposed LFAEC

4 Experimental Setup

4.1 Preprocessing and Mapping

The preprocessing of data using MEDA technique in which the raw data is scaled, normalized, and formulated as frames of data before applying it into the training algorithm. The principal component analysis extracts the agent space vectors present in the raw data. The singular vector decomposition is a unique way of expressing complex patterns of data in which the Smart City data set dynamically changes its values.

4.2 Multivariate Exploratory Data Analysis

The fusion of APCA and SVD algorithm is formulated to provide the dimensionality reduction of a raw dataset by altering the values to data frames. APCA is the enhanced version of the PCA algorithm. The input data contain 42 rows of patterns with columns selected as attributes from numerical columns from 5 to 32 is considered. The first step in APCA is to estimate each row's mean and standard deviation. The covariance matrix is generated by formulating the standardization matrix z. The resultant matrix K is denoted as the variance. The ensemble process of APCA with SVD is achieved by the satisfactory determinants of the APCA matrix, which equals '0'. Then the matrix is considered for further processing, and the process is repeated for calculating APCA. This data is further visualized before mapping. In the MEDA process, the big data is formulated into frames of analysis data in which each data is trained with the APCA-SVD to extract hidden space vectors.

Fig. 2 shows the proposed work flow with data preparation and classification steps. The novelty here is the lightweight architecture that combines the deep learning and machine learning model, where the impacted parameter detection is helpful to make early prediction and prevention in future work.

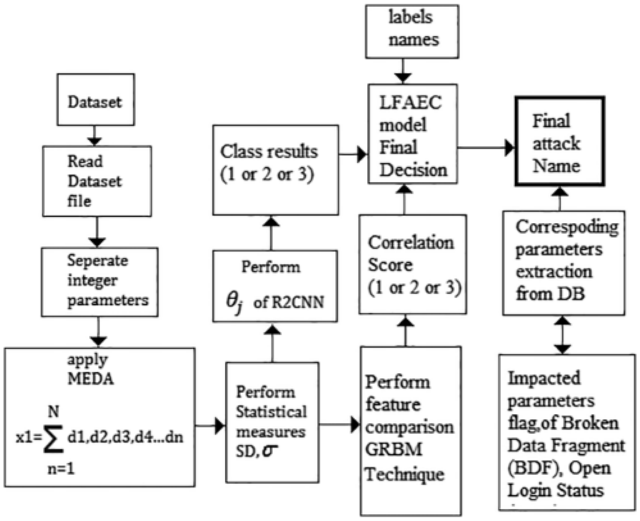

labels names
Dataset
Read Dataset file
Seperate integer parameters
apply MEDA
$x1=\sum_{n=1}^{N} d1,d2,d3,d4...dn$
Class results (1 or 2 or 3)
Perform θ_j of R2CNN
Perform Statistical measures SD, σ
LFAEC model Final Decision
Correlation Score (1 or 2 or 3)
Perform feature comparison GRBM Technique
Final attack Name
Correspoding parameters extraction from DB
Impacted parameters flag,of Broken Data Fragment (BDF), Open Login Status

Figure 2: Flow of proposed study

4.3 Late Fusion Aggregation (LFA)

Late fusion technique enabled the deep learning network with regression model is evaluated. The regression model reflects the correlation results with the feature mapping values from the eigenvectors.

Algorithm: *Cascaded R2CNN*

```
Get input data(Lysis_dataset)
Extract Features X=MEDA (data)
Compute p1=Gradient Regression(X-train,X_test);
Compute p2=R2CNN(X-train,X_test)
Compute LFAEC(p1,p2);
p3=match_score, Repeat loops until Err<0;
Call parameters (max (match data))
Display parameters p1,p2,p3
Repeat all steps.
End Loop
```

The deep recurrent Convolution neural network is adaptively arranged in a cascade way to improve the weights and evaluate the metric model with tuned parameters. The metric score is matched with the presented model with appropriate impact factors according to a display of equivalent inputs for organizing the final decision.

4.4 Late Fusion Aggregation Enabled Cyber Net (LFAEC)

The proposed model is derived from the novel architecture by considering the similarity issues in decision-making. The Novel Late Fusion Aggregation Enabled Cyber-Net (LFEAC) is formulated. The deep network is created using R2CNN architecture. The input data is tuned in such a way it adapts the analysis window of R2CNN. The novel structure is created through the fusion of feature vectors. The late fusion normally happens at the decision stage. The decision stage makes a reliable outcome based on the similarity of the regression result and the R2CNN results. In case of no correlation, the iteration is repeated to make a maximum analysis count.

Let's consider the input dataset matrix to be; input dataset be $\mathrm{I} = \begin{cases} x_1 & x_2 \ldots & x_n \\ y_1 & y_2 \ldots & y_n \\ z_1 & z_2 \ldots & z_n \end{cases}.$

The foremost step of the MEDA process is to find the Standard Deviations (SD) of the given input data. The data is formulated as a matrix that holds the rows of recorded information. Each *ith*row controls the current instant of recorded data of the cyber-physical system.

The standard deviation of I is given by,

$$\sigma = \sqrt{\frac{\sum (xi - \mu)^2}{n}} \tag{1}$$

The MEDA feature derived from transformed values derived from the APCA (after SVD) satisfies the '0' equivalents.

The standardization of the I matrix is denoted by Z, and is given by the formula below.

$$Z = \frac{I - mean\,(I)}{\sigma} \tag{2}$$

Further, the covariance matrix C,

$$C = \begin{bmatrix} Cov\left(z_{1,1}\right) & Cov\left(z_{1,2}\right) & \dots & Cov\left(z_{i,n}\right) \\ Cov\left(z_{2,1}\right) & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ Cov\left(z_{i,n}\right) & \dots & \dots & Cov\left(z_{n,n}\right) \end{bmatrix} \tag{3}$$

where,

$Cov\left(z_{1,1}\right) = s\left(I\right)$

$\mathrm{s}^{n} = \dfrac{\Sigma\left(\mathrm{x_i} - \mathrm{x}\right) n^2}{\mathrm{n} - 1}$

$\mathrm{s}^{n} = Sample\ varience\ \mathrm{x_i} = The\ value\ of\ one\ observations$

$\mathrm{x} = The\ value\ of\ all\ observations$

$n = number\ of\ observations$

where, $\mathrm{x}, y, z \rightarrow Attributes\ selected\ from\ MEDA$

$$K = s\left(I\right) \tag{4}$$

The ensemble operation of APCA and SVD happens,

When SVD satisfies the equation by taking determinants to the variable matrix K,

$\left|k\right| = 0$

$Where\ SVD\left(K\right) = M = u\Sigma v^{t}$

$u \rightarrow Left\ singular\ vector\ matrix$

$v^{t} \rightarrow Right\ singular\ vector\ matrix$

If $\left|k\right| = 0$, then

$$\mathrm{APCA\ (I) = P} = \lambda K \tag{5}$$

As discussed in Section 3, the APCA matrix is further divided into the training set and testing set for regression analysis and R2CNN analysis.

$P \rightarrow denotes\ the\ Tranformed\ original\ inputs$

$\theta_j \rightarrow P \rightarrow atjrows$

Further, the regression framework is derived by the formula below,

$$\theta_j = \theta_j - \alpha \frac{1}{m} \sum\nolimits_{i=1}^{m} x\left(h_\theta\left(x^i - y^i\right)\right) x_j^i \tag{6}$$

$where\ m \rightarrow Number\ of\ Training\ samples;$

$n \rightarrow Number\ of\ features$

The Recurrent CNN weights are derived by the formula below, focusing on the fusion of LFAEC.

$$\theta_k = \frac{\{x_i y_i z_i \ldots n\}_j - 2p - K}{s} + 1 \tag{7}$$

$$p \rightarrow \textit{Convolution Padding Size},\ K \rightarrow \textit{Kernel Size},$$

$$s \rightarrow \textit{Stride Size},\ \overset{0}{\theta}_{k} \rightarrow \textit{Output features}$$

Late fusion aggregation ends after the j^th loop ends, then k^th row of data ends, and the maximum correlation of the decision given by θ_j, θ_k.

$$\text{LFAEC}\left(\text{inp}_{\text{data}}\right) \rightarrow \max \sum\nolimits_{j=1,k=1}^{m} x\left(\theta_j + \theta_k\right) \tag{8}$$

From the above expressions, it is evident that the proposed LFAEC depends on the regression result θ_j and R2CNN results with θ_k obtained for the given training input $x_i y_i z_i \ldots n$. These attributes are considered for analysis till the end of the dataset. Each frame out of 1000 samples has been fetched at every iteration.

5 Discussion

5.1 Preprocessing MEDA Process

Fig. 3 shows the MEDA process, which contains the Advanced PCS component extract from the given data. The eigenspace values are the unique points captured from the given raw data. Fig. 4 shows the result of SVD, which indicates maximum values of '0' in the output when considering 1000 samples. If the maximum occurrence of SVD is '0', then APCA will send out the eigenvalues for evaluation.

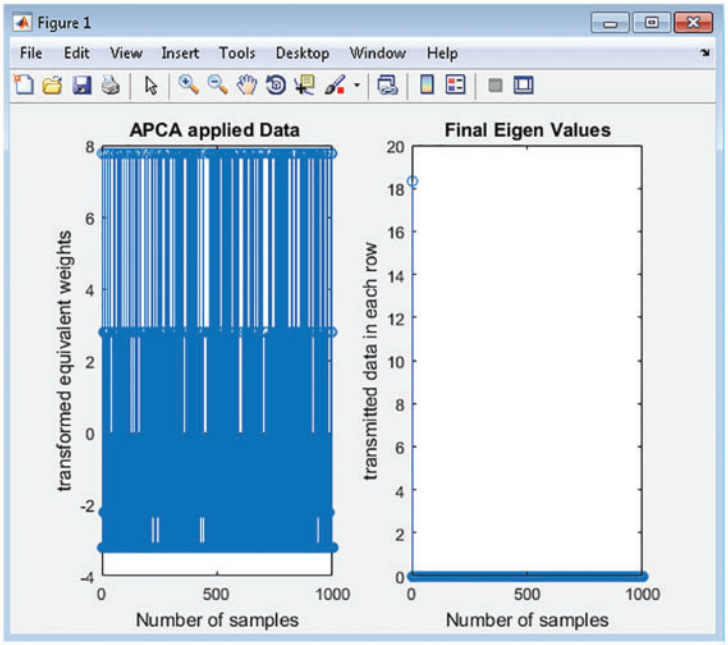


Figure 3: Application of advanced PCA for preprocess

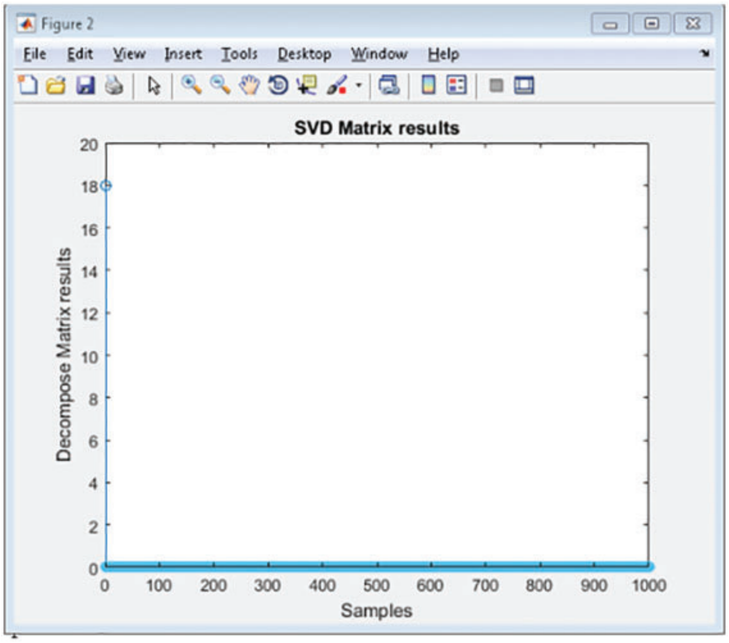


Figure 4: Application of SVD for preprocess

Fig. 5 shows the feature mappings of different inputs along with the APCA model, which in turn shows the status output of each currently connected system. The status is highlighted as running or closed, which depends on the dataset's flag.

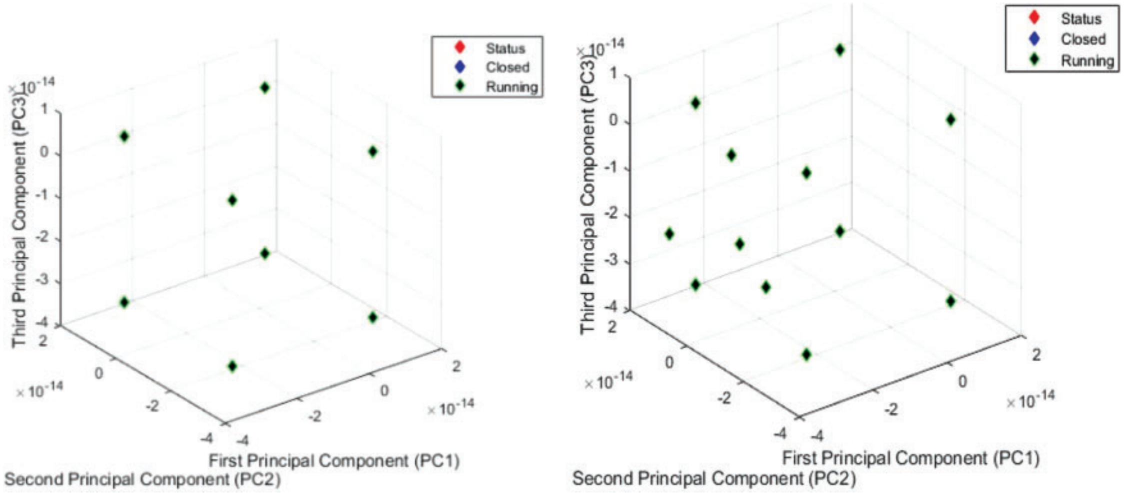


Figure 5: (Continued)

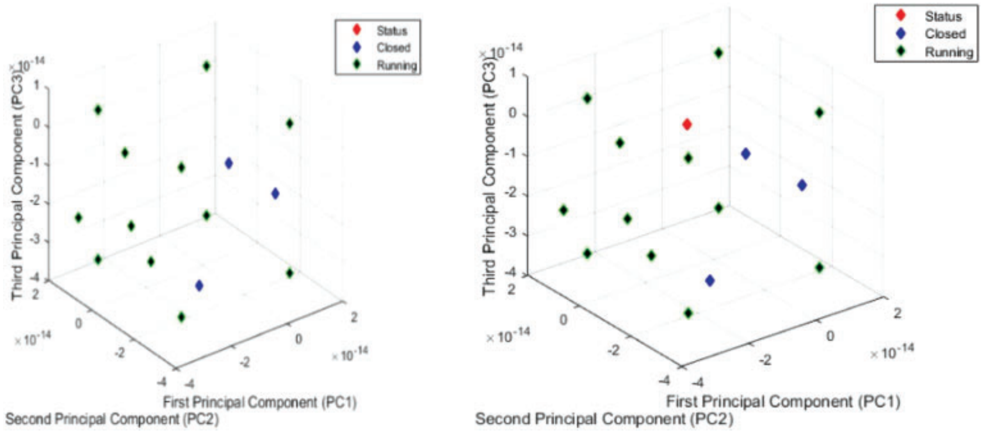


Figure 5: Feature mappings of different inputs

Fig. 6 shows the feature mapping of test input patterns extracted using the MEDA method. The initial analysis visualization enables the system to show the status of each device connected to the network. The indication of running application closed application and status indication is given in the eigen space. The first component, PC1, to the second component, PC2, and the evaluation of the third component, PC3, are mapped. The Eigen data of APCA space is modeled concerning the given frame of 1000 samples.

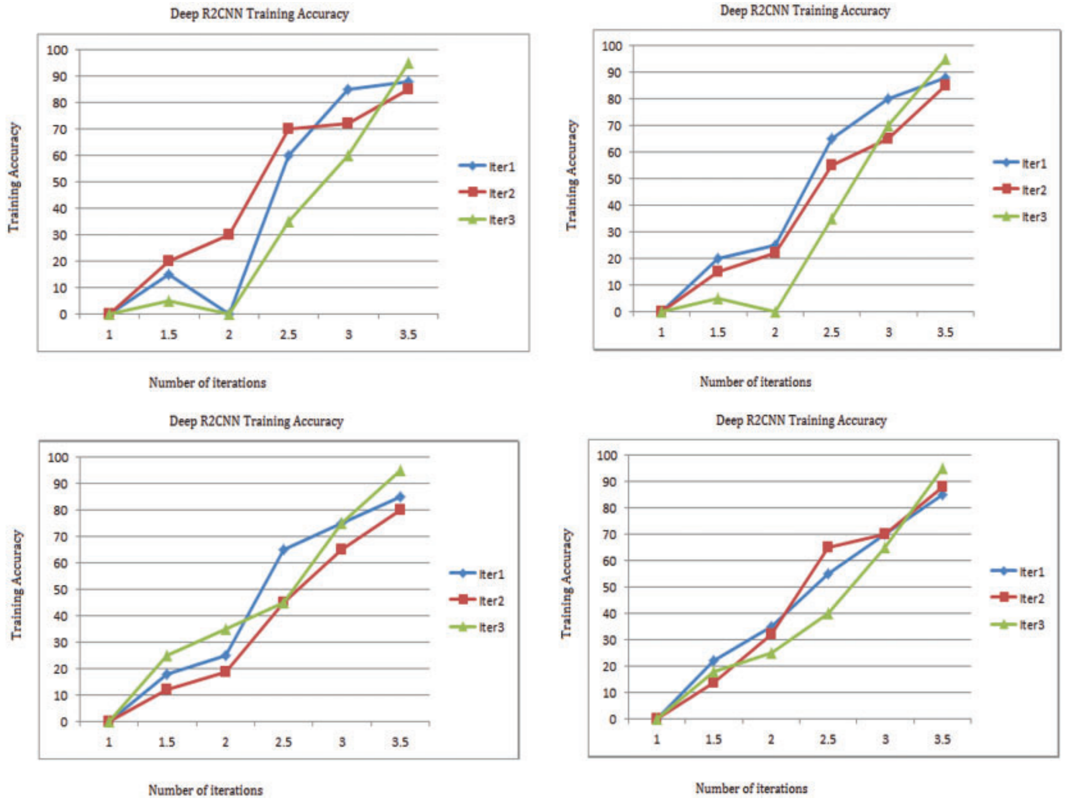


Figure 6: Training accuracy of R2CNN model for various test inputs

5.2 Training Accuracy of Proposed R2CNN

Fig. 6 shows the training accuracy obtained in the R2CNN model with 100 iterations. The Stochastic Gradient Descent with Momentum (SGDM) is the training optimizer with a learning rate of 0.1 at every five epochs, a mini-batch size of 64, Gradient Threshold Method is '***l2norm***'. Validation metrics are included as a parameter during training if a network contains batch normalization layers. The mean and variance statistics used for batch normalization can be different after training completes. The network finally evaluated is displayed after the last iteration. More training epochs improve the accuracy in the case of complex combinations.

Fig. 7 shows the testing accuracy of R2CNN to the iterative analysis of training data given. The resultant classifies the input test pattern shown in the Fig. 8 which indicates the gradient-boosted regression results. This shows the predicted and input feature given, showing the reduced number of samples. The prediction is validated if the count is more than 50% of the input data. Fig. 9 shows the comparative results of GRBM to the direct method that uses the regression by calculating the mean directly and the Inbuilt method that uses the MATLAB fit function to find the separation of given test data and its associated relativity. The regression analysis considers Mean Square Error (MSE) for validation. If MSE is founded less than '1', then the system is considered better performing since MSE needs to be lower in number. If the MSE is '0', then the system is perfect.

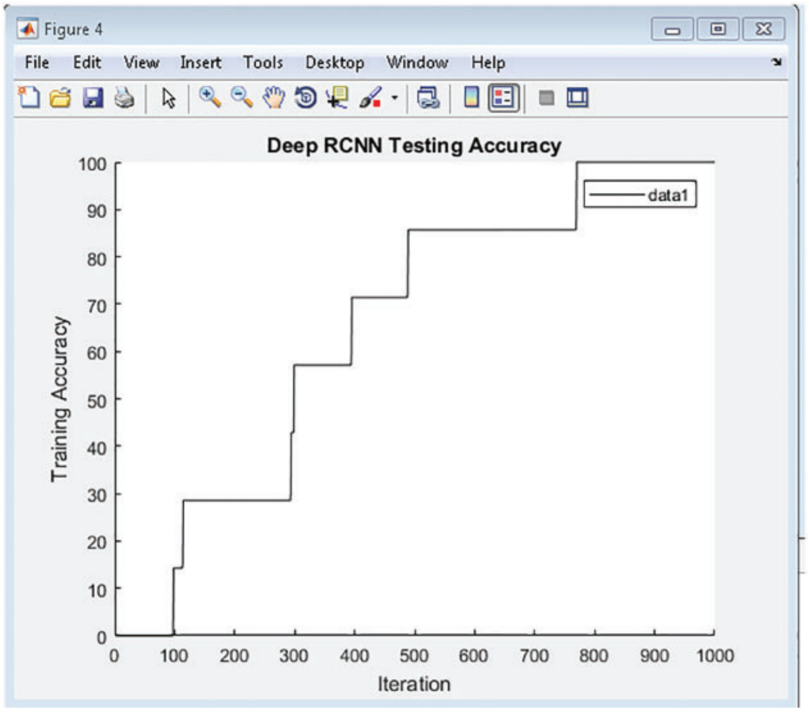
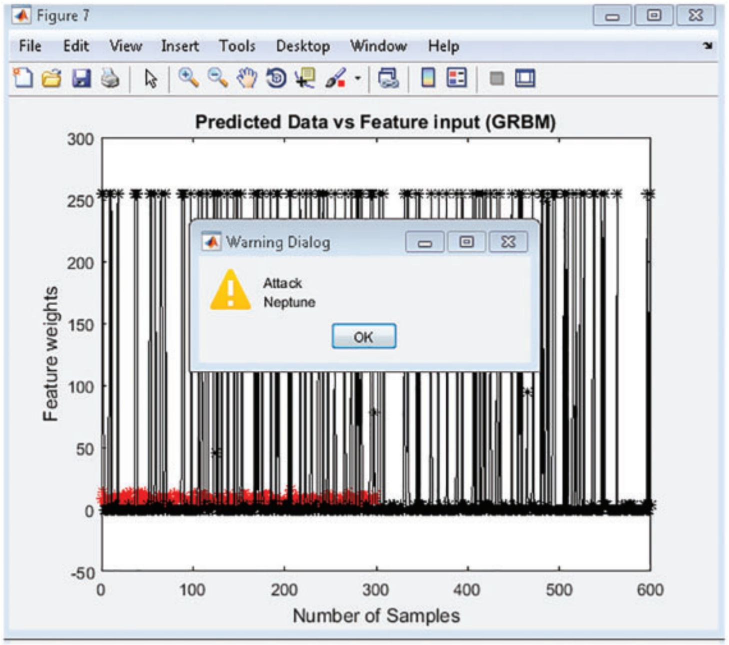


Figure 7: Testing accuracy of proposed model for 1000 epochs



Attack
Neptune

Figure 8: Predicted results *vs.* equivalent feature

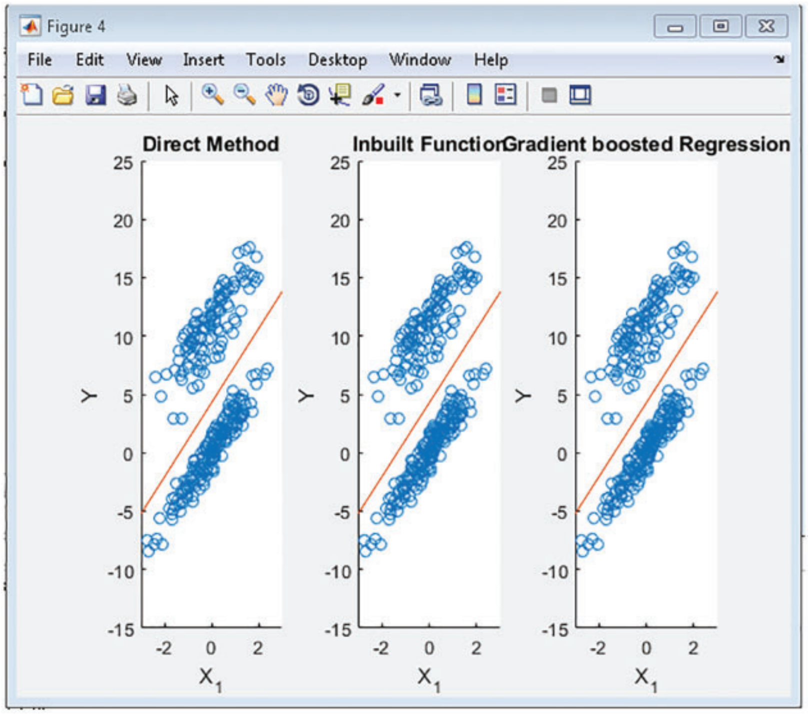


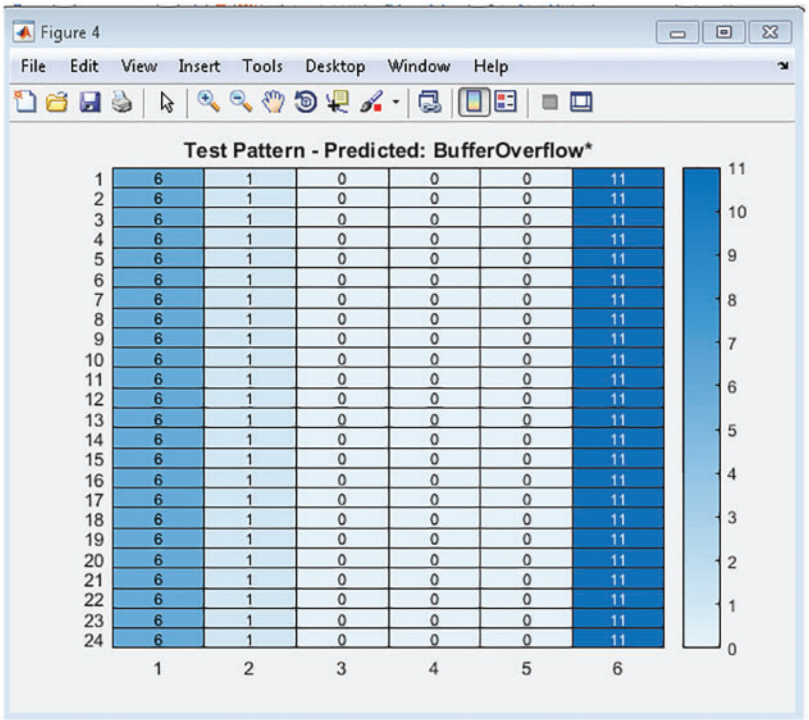
Figure 4
File Edit View Insert Tools Desktop Window Help
Direct Method
Inbuilt Functior
Gradient boosted Regression
Y
25
20
15
10
5
0
-5
-10
-15
-2
0
2
X1

Figure 9: Comparative results of GRBM with respect to direct method and inbuilt conventional method

5.3 LFAEC Decision Making

LFAEC obtains the decision-making derived from the accumulated decision considerations that strengthen the attack scenario. From GRBM, the MSE and Regression factor are considered for decision-making, and the other decision support is gathered from R2CNN. If the prediction is good, then the accuracy is better. The final decision is made, and the R2CNN accuracy is founded to be more than 75%.

Through the complete training data of 25193×42, the test pattern of 2310×42 with features 23103×1 is fetched to the network, and the test pattern impacted for the prediction may occur anywhere. The sample test pattern that impacts the prediction at R2CNN is highlighted here in Fig. 10.



	1	2	3	4	5	6
1	6	1	0	0	0	11
2	6	1	0	0	0	11
3	6	1	0	0	0	11
4	6	1	0	0	0	11
5	6	1	0	0	0	11
6	6	1	0	0	0	11
7	6	1	0	0	0	11
8	6	1	0	0	0	11
9	6	1	0	0	0	11
10	6	1	0	0	0	11
11	6	1	0	0	0	11
12	6	1	0	0	0	11
13	6	1	0	0	0	11
14	6	1	0	0	0	11
15	6	1	0	0	0	11
16	6	1	0	0	0	11
17	6	1	0	0	0	11
18	6	1	0	0	0	11
19	6	1	0	0	0	11
20	6	1	0	0	0	11
21	6	1	0	0	0	11
22	6	1	0	0	0	11
23	6	1	0	0	0	11
24	6	1	0	0	0	11

Figure 10: Sample test pattern detected out for attack name Buffer_Over_Flow

5.4 Performance Analysis

The statistical performance of the system is derived from formulating the confusion matrix at the R2CNN model and prediction with the actual result from the GRB model etc. The utilized formulas are given in Table 1.

Table 2 Discuss the various attack detected with impacted parameters such as Source Time (Ts), Destination Time (Td), Source IP (Sip), Destination IP (Dip), Trigger Flag (Trf), and Location data (Loc), etc.

Fig. 11 compares parameters such as Accuracy, Precision, Recall, F1Score, Sensitivity, and Specificity of existing systems.

Table 1: Quantitative measures

Accuracy $=$ TP $+$ TN/TP $+$ TN $+$ FP $+$ FN
Precision $=$ TP/TP $+$ FP
Recall $=$ TP/TP $+$ FN
F1Score $= 2 \times$ Precision $\times$ Recall/(Precision $+$ Recall)
Sensitivity $=$ TP/TP $+$ FN
Specificity $=$ TN/TP $+$ FN

Table 2: Detected attacks and impact parameters

S. No	Attack name	Impact attributes	Accuracy	Loss
1	UnknownAPK	Ts, Td, Sip, Dip	99.25	0.05
2	Password_Guess	Ts, Td, Sip, Dip	99.25	0.02
3	Smurf	Ts, Td, Sip, Dip	99.24	0.03
4	TearDrops	Sip	99.15	0.04
5	Probes	Trf, Loc	99.27	0.02

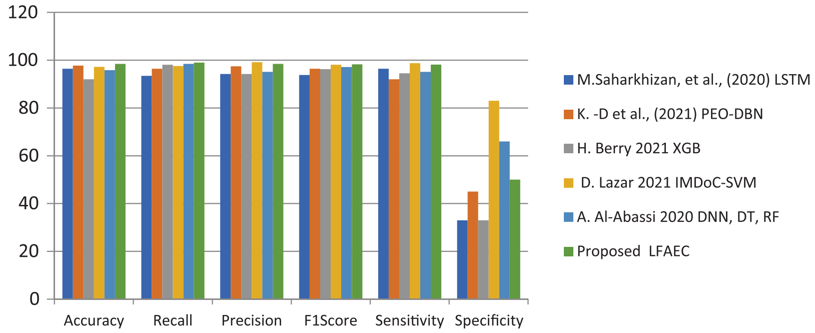


Figure 11: Comparison of accuracy, recall, precision, F1Score, sensitivity, and specificity of existing systems

Table 3 Compares various parameters impacted by cyber-attacks. The threat loops disrupt cybersecurity through some of the impact parameters in the pattern of device communication. Table 4 shows the consideration of Broken Data Fragment (BDF), Open Login Status (OLC), Suspicious Number of Connected PC (SNC), Weak (Admin) Root Shell (WR), Destination Host Count (DHC), Switch user attempts (SU) are utilized as impact parameters. Fig. 12 shows the most impacted parameter when the attack is generated, and the maximum range of certain parameters is depicted. Fig. 12 shows parameters impacted during normal operations. The LFA elapsed time taken to complete the process is 3.124 s. The normal operation is held, if another suspicious system connectivity is avoided.

Table 3: Comparison of existing parameters with the proposed LFAEC model

References	Method	Accuracy	Recall	Precision	F1Score	Sensitivity	Specificity
[12]	LSTM	96.45	93.45	94.205	93.812	96.45	33
[24]	PEO-DBN	97.75	96.45	97.45	96.45	92	45
[13]	XGB	92	98.125	94.25	96.25	94.5	33
[17]	IMDoC-SVM	97.2	97.554	99.15	98.12	98.745	83
[18]	DNN, DT, RF	95.86	98.47	95.12	97.15	95.148	66
Proposed	LFAEC	98.45	99	98.45	98.25	98.14	50

Table 4: Comparison of various parameters impacted on cyber attacks

Parameters impacted	SMURF	Neptune	GuessPw	Buffer overload	IpSweep	Nmap	Normal
BDF	3	4	5	6	7	0	9
OLS	1	0	1	1	0	1	0
SNC	38	0	0	0	0	884	0
WR	1	0	0	0	0	1	0
DHC	3	4	5	6	7	0	9
SU	0	0	0	0	0	2	0
LFAEC-time	7.8996	8.7752	0.32524	0.019201	4.7903	4.2036	3.124

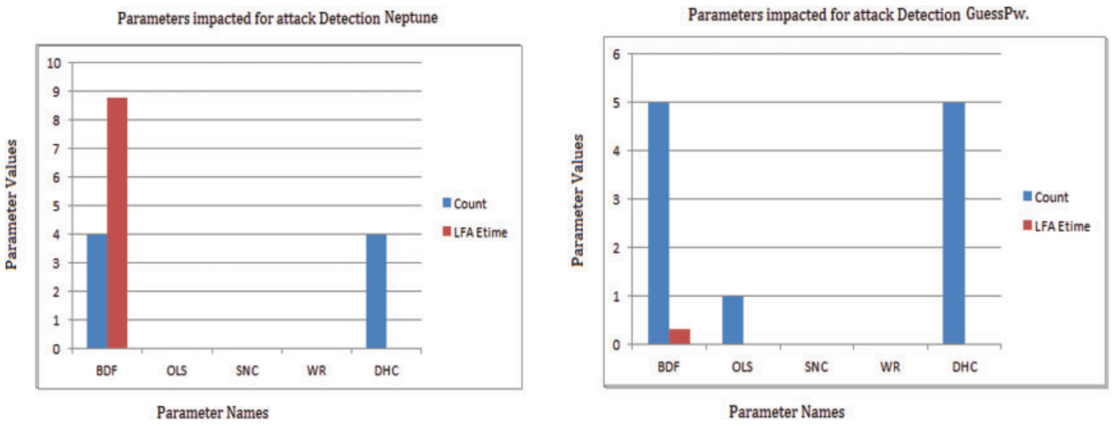


Figure 12: Impact parameters and equivalent values of the maximum range

5.5 Impact Parameters on Various Attacks

Fig. 12 shows the impacted parameters, and their equivalent maximum range is detected. During the detection of the attack scenario, the highly impacted parametric values dominate the results. The dataset is highly imbalanced and large. Accurate formulation of sensitive parameters is shown in Fig. 12, helpful to make accurate decisions on attacks or normal [29–32].

5.6 *Late Fusion Aggregation Elapsed Time*

Fig. 13 shows that the Late Fusion Aggregation (LFA) technique represents the elapsed time on finding the attacks. Many existing models are discussed in Section 4, A deep belief network which is a robust and slow model. The drawback of detection models in existing works are slow prediction time. In this context, a minimum of 0.32 s for the Guess_Pw attack and 0.0192 for Buffer_Over_Load are detected. The slower detection of attacks is not helpful for the robust model. The resilience in detection time directly impacts the flow of attack apps hitting the channels. A maximum of 8.7752 s is required to detect a Neptune attack. The main challenge of the present system is the evaluation of large datasets with limited infrastructure. The smart city dataset is one of the largest scopes that collects the accumulated data of daily time stamps, source IP, destination IP, system ID, location, and even more, arranged in large, scattered mappings. For the present analysis, the focus is on multi-modality issue-solving and processing delay reduction. The dataset is preprocessed initially. Exploratory data analysis helps organize the data properly. Further, the Smart city datasets are collectively organized with a deep feature selection method, and predictions are focused on Generative Aggressive Network (GAN) for better evaluation.

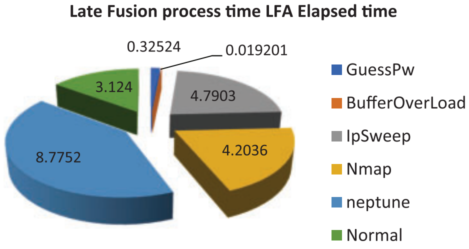


Figure 13: LFA processing time comparison

6 Conclusion

Cyber-attack detection and counteraction are quite common in internet-oriented service platforms. The proposed framework is focused on implementing a robust procedure to detect the most important reasons behind cybersecurity malfunctions in IoT networks. The framework comprises Advanced Principal Component Analysis (APCA) and Singular Vector Decomposition (SVD) for dimensionality reduction. The Late Fusion (LF) technique is evaluated for aggregating the recurrent 2 neural network model and gradient regression model to provide validated results in final decision modeling. The proposed forecast model accomplishes almost 98.45% accuracy towards the static dataset obtained from LYSIS. The processing methods take 10.774826 s. The presented model detects the cyber-attacks such as Unknown_APK, Password_Guess, Smurf, Tear_Drops, and Probes. Since the proposed work is implemented on the dynamic data collected from a smart city's specific location, it has various locations and its parametric information are included and tested. The limitation persists with the static data. An exploration of global analysis is recommended. Further, the presented model needs to be improved through Generalized Aggressive Network (GAN) for adapting the learning process, incorporating multiple deep learning architectures to be validated comparatively.

Acknowledgement: The Authors would like to express their gratitude to Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (Deemed to be University) for providing the opportunities and facilities to conduct the research study.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] C. Konstantinou, M. Maniatakos, F. Saqib, S. Hu and J. Plusquellic, “Cyber-physical systems: A security perspective,” in *20th IEEE European Test Symp. (ETS)*, Cluj-Napoca, Romania, pp. 1–8, 2015.

[2] Y. Joo, Z. Qu and T. Namerikawa, “Resilient control of cyber-physical system using nonlinear encoding signal against system integrity attacks,” *IEEE Transactions on Automatic Control*, vol. 66, no. 9, pp. 4334–4341, 2021.

[3] Y. Zhou, F. R. Yu, J. Chen and Y. Kuo, “Cyber-physical-social systems: A state-of-the-art survey, challenges and opportunities,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 389–425, 2020.

[4] T. Li, B. Chen, L. Yu and W. A. Zhang, “Active security control approach against DoS attacks in cyber-physical systems,” *IEEE Transactions on Automatic Control*, vol. 66, no. 9, pp. 4303–4310, 2021.

[5] H. M. Chung, W. T. Li, C. Yuen, W. H. Chung and Y. Zhang, “Local cyber-physical attack for masking line outage and topology attack in smart grid,” *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4577–4588, July 2019.

[6] C. Yang, Z. Shi, H. Zhang, J. Wu and X. Shi, “Multiple attacks detection in cyber-physical systems using random finite set theory,” *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 4066–4075, 2020.

[7] S. Sodagudi, S. Kumari Kotha and M. David Raju, “Novel approaches to identify and prevent cyber attacks in Web,” in *Int. Conf. on Computing Methodologies and Communication (ICCMC)*, Erode, India, pp. 833–839, 2019.

[8] L. K. Mestha, O. M. Anubi and M. Abbaszadeh, “Cyber-attack detection and accommodation algorithm for energy delivery systems,” in *IEEE Conf. on Control Technology and Applications (CCTA)*, Mauna Lani, HI, pp. 1326–1331, 2017.

[9] M. Saharkhizan, A. Azmoodeh, A. Dehghantanha, K. K. R. Choo and R. M. Parizi, “An ensemble of deep recurrent neural networks for detecting IoT cyber attacks using network traffic,” *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8852–8859, 2020.

[10] H. Berry, M. A. Abdel-Malek and A. S. Ibrahim, “A machine learning approach for combating cyber attacks in self-driving vehicles,” in *Proc. of the Southeastcon*, Atlanta, GA, USA, pp. 1–3, 2021.

[11] Z. N. Zarandi and I. Sharifi, “Detection and identification of cyber-attacks in cyber-physical systems based on machine learning methods,” in *Int. Conf. on Information and Knowledge Technology (IKT)*, Tehran, Iran, pp. 107–112, 2020.

[12] Y. Feng, H. Akiyama, L. Lu and K. Sakurai, “Feature selection for machine learning-based early detection of distributed cyber attacks,” in *IEEE 16th Int. Conf. on Dependable, Autonomic and Secure Computing, 16th Int. Conf. on Pervasive Intelligence and Computing, 4th Int. Conf. on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*, Athens, Greece, pp. 173–180, 2018.

[13] K. Shaukat, S. Luo, S. Chen and D. Liu, “Cyber threat detection using machine learning techniques: A performance evaluation perspective,” in *Int. Conf. on Cyber Warfare and Security (ICCWS)*, Islamabad, Pakistan, pp. 1–6, 2020.

[14] D. Lazar, K. Cohen, A. Freund, A. Bartik and A. Ron, “IMDoC: Identification of malicious domain campaigns via DNS and communicating files,” *IEEE Access*, vol. 9, pp. 45242–45258, 2021.

[15] A. Al-Abassi, H. Karimipour, A. Dehghantanha and R. M. Parizi, “An ensemble deep learning-based cyber-attack detection in industrial control system,” *IEEE Access*, vol. 8, pp. 83965–83973, 2020.

[16] N. Saxena, L. Xiong, V. Chukwuka and S. Grijalva, “Impact evaluation of malicious control commands in cyber-physical smart grids,” *IEEE Transactions on Sustainable Computing*, vol. 6, no. 2, pp. 208–220, 2021.

[17] M. Barbeau, F. Cuppens, N. Cuppens, R. Dagnas and J. Garcia-Alfaro, "Resilience estimation of cyber-physical systems via quantitative metrics," *IEEE Access*, vol. 9, pp. 46462–46475, 2021.

[18] T. Sui, Y. Mo, D. Marelli, X. Sun and M. Fu, "The vulnerability of cyber-physical system under stealthy attacks," *IEEE Transactions on Automatic Control*, vol. 66, no. 2, pp. 637–650, 2021.

[19] M. Keshk, B. Turnbull, E. Sitnikova, D. Vatsalan and N. Moustafa, "Privacy-preserving schemes for safeguarding heterogeneous data sources in cyber-physical systems," *IEEE Access*, vol. 9, pp. 55077–55097, 2021.

[20] R. Heartfield, G. Loukas, A. Bezemskij and E. Panaousis, "Self-configurable cyber-physical intrusion detection for smart homes using reinforcement learning," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1720–1735, 2021.

[21] K. D. Lu, G. Q. Zeng, X. Luo and J. Weng, "Evolutionary deep belief network for cyber-attack detection in industrial automation and control system," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 11, pp. 7618–7627, 2021.

[22] E. Mousavinejad, F. Yang, Q. L. Han, X. Ge and L. Vlacic, "Distributed cyber attacks detection and recovery mechanism for vehicle platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3821–3834, 2020.

[23] C. Dong, H. Wang, D. Ni, Y. Liu and Q. Chen, "Impact evaluation of cyber-attacks on traffic flow of connected and automated vehicles," *IEEE Access*, vol. 8, pp. 86824–86835, 2020.

[24] R. Khilar, K. Mariyappan, M. S. Christo, J. Amutharaj, T. Anitha *et al.*, "Artificial intelligence-based security protocols to resist attacks in internet of things," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1440538, pp. 1–10, 2022. http://doi.org/10.1155/2022/1440538

[25] D. Lee and D. Kundur, "Cyber-attack detection in PMU measurements via the expectation-maximization algorithm," in *IEEE Global Conf. on Signal and Information Processing (GlobalSIP)*, Atlanta, GA, pp. 223–227, 2014.

[26] A. Sivanathan, H. H. Gharakheili and V. Sivaraman, "Detecting behavioral change of IoT devices using clustering-based network traffic modeling," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7295–7309, 2020.

[27] J. B. Karande and S. A. Joshi, "Comprehensive assessment of security attack detection algorithms in internet of things," in *Int. Conf. on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, pp. 1–6, 2018.

[28] U. A. Bhatti, L. Yuan, Z. Yu, J. Li, S. A. Nawaz, *et al.*, "New watermarking algorithm utilizing quaternion Fourier transform with advanced scrambling and secure encryption," *Multimedia Tools and Applications*, vol. 80, no. 9, pp. 13367–13387, 2021. https://doi.org/10.1007/s11042-020-10257-1

[29] D. Sathiya and S. Sheeja "Data delivery and node positioned learning automaton in mobile ad hoc networks," *Journal of Computational Science and Intelligent Technologies*, vol. 3, no. 2, pp. 1–14, 2022. https://doi.org/10.53409/MNAA/JCSIT/e202203020114

[30] K. Swathine and N. Sumathi, "A meta-heuristic approach based on adaptive optimization for tracking software requirements," *Journal of Computational Science and Intelligent Technologies*, vol. 3, no. 2, pp. 15–30, 2022. https://doi.org/10.53409/MNAA/JCSIT/e202203021530

[31] S. Manimurugan, A. Majdi, M. Mohammed, C. Narmatha and R. Varatharajan, "Intrusion detection in networks using crow search optimization algorithm with adaptive neuro-fuzzy inference system," *Microprocessors and Microsystems*, vol. 79, no. 1, p.103261, 2020.

[32] S. Manimurugan, S. Almutairi, M. M. Aborokbah, C. Narmatha and S. Ganesan, *et al.*, " Two-stage classification model for the prediction of heart disease using IoMT and artificial intelligence," *Sensors*, vol. 22, no. 2, p. 476, 2022. https://doi.org/10.3390/s22020476