XGBoost Regression Classifier (XRC) Model for Cyber Attack Detection and Classification Using Inception V4

K. M. Karthick Raghunath[1], V. Vinoth Kumar[2], Muthukumaran Venkatesan[3], Krishna Kant Singh[2,*], T. R. Mahesh[2] and Akansha Singh[4]

[1]*Department of Computer Science & Engineering, MVJ College of Engineering, Bangalore, India*
[2]*Department of Computer Science and Engineering, Jain (Deemed to be University), Bangalore, India*
[3]*Department of Mathematics, School of Applied sciences, REVA University, Bangalore, India*
[4]*School of Computer Science Engineering and Technology, Bennett University, India*
E-mail: raguaut@gmail.com; drvinothkumar03@gmail.com; muthu.v2404@gmail.com; krishnaiitr2011@gmail.com; trmahesh.1978@gmail.com; akanshasing@gmail.com
[*]*Corresponding Author*

Received 12 December 2021; Accepted 22 February 2022; Publication 18 April 2022

Abstract

Massive reliance on practical systems has resulted in several security concerns. The ability to identify anomalies is a critical safety feature enabled by anomaly diagnostic techniques. The construction of a data system faces a significant issue in cyber security. Because of the exploitation of valuable data, cybersecurity impacts the privacy of such data. Attack incidents

Journal of Web Engineering, Vol. 21_4, 1295–1322.
doi: 10.13052/jwe1540-9589.21413

must be examined using an appropriate analytics approach in elevating the safety level. Design of advanced analytical, conceptual model creation gives practical guidance and prioritizes threats/attacks across the network system. There is now substantial effectiveness in attack categorization, and evaluation through Convolution Neural Network (CNN) based classifiers. In light of the drawbacks of previous approaches, this research proposes an approach relying on the Deep Learning (DL) strategies for cyberattacks detection and categorization in the context of cyberspace incidents. Likewise, this article presents an XGBoost Regression Classifier (XRC) using Inception V4 to address those restrictions. XGBoost refers to Extreme Gradient Boosting, a decentralized gradient-boosted decision tree (GBDT) supervised learning framework that is robust and can be used in a decentralized context. XGBoost is a well-known machine learning technique because of its ability to produce outstanding accuracy. The concepts of both XGBoost and Regression classifiers are integrated and represented as a suggested hybridized classifier, which is implemented in Inception V4 to further train and test the model. The proposed XRC categorizes and forecasts several common types of network cyberattacks that includes Distributed Denial of Service (DDoS), Phishing, Cross-site Scripting (CS), Internet of Things (IoT). The sigmoidal function is used as a supportive activator to the hybridized classifier to lower the erroneous ratio and increase the effectiveness. Research shows that training and testing errors were substantially decreased when using XRC. In 9 out of 13 instances, over 97% of threats are detected by the XRC, and over 75% of threats are detected in its most challenging datasets.

Keywords: Cybersecurity, XGBoost regression classifier (XRC), inception V4, hybridized classifier, error rate.

1 Introduction

The creation of web apps plays a crucial part in the day-to-day routines of human-computer interaction existence. Integrated network infrastructure offers a more comprehensive platform for critical data storage and computation. Since sensitive information is being processed, network interaction must be protected with suitable cybersecurity [1, 7]. Various security measures have been implemented to safeguard web-based networking against cyberattacks, including virus protection, user security mechanisms, firewalls, and authorization approaches. Security mechanisms for communications infrastructure have been created; however, many fail to meet their stated objectives,

resulting in an increased risk [2]. According to a study provided by the United States in 2014, there were several vulnerabilities. Russian cyber-attacks of varying intensity were also reported in 2007 [3]. Increasing dataset dimensions and dynamic workspace, along with more extensive sampling, have combined study areas in recent times to provide a more effective Intrusion Detection System (IDS). An IDS searches for and classifies various features in the traffic information flow to distinguish between malicious and benign intrusions [4–6]. There was a role for the cyber defence infrastructure in the vulnerability assessment for future decision-making. On utilizing such a technique, risk in the workplace is systematically assessed and managed. We have previously stated that cyber security is a crucial problem that instantaneously impacts mechanisms and dynamic assessments [7–10]. Experts can make quick judgments depending on network assessment in a stable cyber environment. However, a static aspect is required to maintain the entire platform stable, even though this has necessitated a detailed study and evaluation of cybersecurity issues [11–14].

In a highly complex and escalating adversary context, the cyberspace infrastructure must establish efficient methodologies and examine cybersecurity risk mitigation. Additionally, the design technique must be accompanied by privacy and protection requirements. In the conventional cyber context, risk mitigation in web engineering is always managed with Global Regulations [15–21]. Several web engineering standards have been developed to streamline and govern technological operations. While requiring sound web engineering with incorporated security and safety considerations, these guidelines don't enable autonomous unit protection for managing purposes. Cybersecurity focused on web engineering establishes a unique defence genre for organisations to meet those constraints.

1.1 Motivation of the Proposed Work

The significance of data security is rising as the web engineering domain expands. However, owing to the high incidence of inaccuracy, this does not give a substantial remedy for cybersecurity threats [14]. Hybridized Deep Learning (DL) has recently attracted a lot of attention, especially in the sphere of cybersecurity. Using these strategies, training and computation capacity may be applied to practical and theoretical situations. Cyberattacks have been studied extensively [22–25]; however, the error rate is substantially greater than other fields of study. This concern led to the use of the XGBoost Regression Classifier (XRC) to recognize and segment cyber-attack incidents in this

work [35]. For better classification, the proposed XGBoost and Regression classifiers were combined. Even though the proposed strategies framework has its own unique Application Programming Interfaces, we planned to utilize it via the scikit-learn wrapper classes that further includes XGBClassifier and XGBRegressor. For data preparation and model evaluation, we will be able to use the whole scikit-learn ML package. The sigmoidal activity function is used in the integrated classification model to lower the error margin of the classifier. Improved identification and categorization of attack instances are achieved using the proposed XRC in Inception V4. The simulation outcomes showed that the suggested XRC significantly decreases the detection error margin.

The following are the study's objectives, which were prompted by the following observations:

i. To develop an XGBoost Regression Classifier (XRC) for the recognition and segmentation of cyberattacks incidents.
ii. An evaluation relying on root mean square error (RMSE), error rate, and receiver operating characteristics (ROC) curve are estimated using other criteria such as precision, accuracy, recall, and F1-Score.
iii. An evaluation relying on root mean square error (RMSE), error rate, and receiver operating characteristics (ROC) curve are estimated using other criteria such as precision, accuracy, recall, and F1-Score.

This research article is outlining the entire proposed process as follows: Section 1 delineates generalized concepts of cyber attacks and their associated counter-approaches. Section 2 comprises the relevant works and highlights their drawbacks. Section 3 describes the essential preliminaries, dataset followed by procedures of the proposed XRC with Inception V4. Section 4 presents the experimental outcomes and analyzed along with existing approaches. Ultimately, Section 5 concludes the proposed work in positive notes.

2 Related Works

This discussion section will go through several prominent methods that previous researchers have done.

To detect intrusions, [2] recommends a statistical approach. First, a comprehensive data portfolio is built (task) to characterise a certain topic (web user) or an item. For each category, a set of measurements have been established. It is the goal of the indicators' stochastic classifiers to identify

breaches. A statistical analysis of n-grams is used in [14] to identify attacks in the host machine. Instead of relying on data sets, [26] employ active network data to build profile information, which improves the study on web-based intrusion detection techniques.

Complicated criteria and mathematical analysis are proposed by certain scholars. Researchers examined assaults on routing algorithms by assessing the incidence of individual events connected to the algorithm and proposing a criterion for computing the correlation between actual and predicted distributions of events. The measurement is expected to follow a chi-square pattern. An incidence probability pattern is extracted, and then its chi-square measurement is accomplished from a predicted probability vector. The range is expected to follow a normal Gaussian distribution. The hyperbolic curve is used to simulate certain intrinsic properties of IP data stream proposed [13]. Since we've covered static data schemas, we'll move on to dynamic data forms. It's also possible to take advantage of dynamic data representations. There is an assumption that the initial gradient of the proportion of recorded incidents in a temporal frame follows the Poisson process, through which Kolmogorov analytical measures may be recovered to evaluate the difference among observed systems and predefined patterns [27].

A Stochastic random variable is utilized in [28] to depict a time-ordered series of incidents. Regular network activity and attacks may be distinguished using the conditional distribution of a certain sequence of incidents. It has recently been applied in attack detection methods relying on server data records [29] to utilize the hidden Markov prototype. There've been significant advances in detecting cyberattacks based on ML and pattern recognizing approaches. Learning methods such as guided and unguided instruction are also used. A Neural Network (NN) proposed in [30] distinguishes attacks and normal actions. Researchers merge the encoding of category elements and alphanumeric fields to map the network information to a NN. Hierarchical NNs are proposed in [24] as a technique for vulnerability scanning. Evolutionary neural networks (EVNNs) are used in [36] to identify unauthorized intrusions. Fuzzy-based K-means and adaptive classification are used to identify cyberattacks in [31]. In addition, cyberattacks can be detected using the progressive clustering approach proposed in [32], which enhances the K-means approach. Only publicized incursions can be detected using supervised training strategies for detecting attacks. However, the invasions that haven't yet already been trained may be detected using unsupervised training strategies. For example, the K-means technique and the self-organizing

Table 1 Comparison of Existing Approaches with prominent factors

References	Approach	Issues	Performance
[2, 14]	Statistical Analysis	high bias, low variance	Accuracy is low
[28]	Stochastic random variable	low variance, over fitting	Lower than XGBoost
[29]	Hidden Markov Prototype	Avoid Over-fitting	Lower than XGBoost
[34]	SVM	high bias, low variance	Less prone to Over-fitting issues
[3, 9, 31, 32]	K-Means Algorithm	high bias, low variance	Accuracy is low

classification model (SOM) strategy are instances of unsupervised classification for vulnerability scanning [3, 9]. The work in [33] utilized Support Vector Machines (SVMs) to discriminate among typical network activities and attacks, as well as to find additional characteristics for vulnerability scanning. The work in [34] suggests ArraySVM and TreeSVM as a solution to the inefficiency of minimal serial level optimizing computation when dealing with vast sets of input samples during malware detection. The article from [25] presents a strategic e-learning form of SVMs, especially for actual threat detection dependent on an enhanced text classification model.

Table 1 represents the comparison of vital factors of existing approaches. In light of the drawbacks of previous approaches, this research proposes an approach relying on the Deep Learning (DL) strategies for cyberattacks detection and categorization in the context of cyberspace incidents. A comprehensive review testbed was used to determine the necessary technique for detecting cyber-attacks in web engineering through analysis. XRC for cyber security is the proposed approach. The suggested model incorporates a nonlinear system for the Inception V4 framework to train and test the model for efficient detection and classification of cyberattacks. The suggested XRC evaluation is then used to quantify the experimental values associated with cybersecurity estimation and prognosis in web engineering.

3 Preliminaries

This section explains a sample dataset used to classify and identify attacks. For the purpose of identifying cyber attacks in the network, the cyber datasets

Table 2 CSE-CIC-IDS2018 dataset attribute selection features

Attributes	
Source IP	Bytes Utilized
Source Port	Packets Utilized
Destination IP	Flags
Destination Port	Class_Type
Protocol ID	Attack_Type
Date first seen	Attack_ID
Duration	Attack_Description

EMBER [25], CSE-CIC-IDS 2018 [26], and the Unified Host and Network (UHN) [25] Data Set are employed.

3.1 Communications Security Establishment (CSE) & the Canadian Institute for Cybersecurity (CIC) (CSE-CIC-IDS 2018)

The CSE-CIC-IDS2018 datasets are multidimensional, comparable to the CICIDS2017, KDD Cup 99, and UNSW-NB15 datasets. CSE-CIC-IDS2018 was designed to train classification and prediction models for web vulnerability tracking and further study oddity detection using various ML techniques.

CSE-CIC-IDS2018 comprises approximately 16,000,000 incidences gathered during a ten-day timeframe. This was the recently updated cyber security dataset, which is significant, publicly accessible, and encompasses a broader variety of attack variants. The programme generates a CSV file containing six fields identified on every stream, notably Source IP, Destination IP, Flow_ID, protocol utilized, Source, and Destination Port. Each of these has over 80 web traffic characteristics. Some of vital attributes for selected CSE-CIC-IDS2018 dataset were presented in below Table 2.

3.2 Unified Host and Network (UHN) Dataset

The UHN Dataset is a collection of networking and computing activities gathered over almost ninety days from the National Laboratory of Los Alamos enterprise. The information is generated in CSV format that comprises time, timeframe, source and destination device, port, packets, source bytes, procedure, etc. Table 3 represents the essential attributes of the concerned dataset.

Table 3 UHN dataset attribute selection features

Attributes	
Epoch Time	Protocol Number
Duration/event	Attack Class
Source Device	Destination Device
Source Packets	Destination Packets
Source Bytes	Destination Bytes
Source Port	Destination Port

3.3 Endgame Malware BEnchmark for Research (EMBER)

The EMBER dataset is a compilation of features extracted from PE (Portable Executable) files that the researcher uses as a test set. It is a publicly available dataset used to train ML models to instantly identify harmful PE files. The dataset consists of 1.1 million executable files: 900,000 training instances (300,000 benign, 300,000 malicious and 300,000 unclassified) and 200,000 testing data (100,000 for both malicious and benign).

3.4 Dataset Evaluation

Managing cybersecurity in the Web Engineering domain have a problem with increasing computational latency, which results in a decrease in effectiveness and precision. To address this problem with XRC, redundant data will be removed using an optimized subset during the preprocessing stage. Additionally, unnecessary features are excluded from the sample during preprocessing without any adverse effect on the precision or computational overhead. XRC includes feature extraction for data streamlining, dimension reduction, and reduced training duration for large data sets. Following the data set's preprocessing, extraction is used to reduce the data dimension. The selection of refined features lowers duplication while increasing the relevance of data. XRC also includes three separate feature criteria: wrapping, filtering, and integrated model. The suggested XRC is used for categorization learning in this study. Certain criteria restrict the properties of both testing and training datasets of interdependence, entropy, coherence, association and range impact [20]. To identify attributes in the wrapping framework, a predefined learner is employed. In the context of a filtering model applied to extracted features of a trained classifier, several rounds with an adequate standard. The learning and testing rate of proposed system fixed at 0.1 whereas both training and testing proportion of each dataset are fixed as 80:20.

4 XGBOOST (eXtreme Gradient Boosting) Regression Classifier

In general, XGBoost is a strong classifier in ensemble methods used to predict the output efficiently [15]. XGBoost is a very effective technique for developing guided logistic regression. However, the veracity of this proposition may be determined by understanding its optimal solution (XGBoost) and baseline trainees. XGBoost was developed primarily for quickness and reliability via the use of basic principles of gradient-boosting. The sigmoidal equation used in this study for the XCR is as follows,

$$\mathbb{h} = \boldsymbol{sig}\left(\sum_{\boldsymbol{i}} (\boldsymbol{a_i}) \times \boldsymbol{\beta_i H_i}\right) \tag{1}$$

The classifier's performance in detecting the cyberattacks events was improved in this study by combining Logistics Regression (LR) with the XGBoost Classifier, which is referred to as "XCR". Both binary and multi-label categorizations are utilized in LR. In LR, the logistic function is used to forecast the likelihood of matching facts (fitting data). Hence, the function has a value ranging between 0 and 1; if the value exceeds 0.5, immediately presumed to be 0, which is determined using Equation (2).

$$\boldsymbol{H_{\varphi}}(\boldsymbol{a}) = \mathcal{G}\left(\mathbf{1}/\left[\mathbf{1} + \boldsymbol{e}^{-\boldsymbol{\varphi i}}\right]\right) \tag{2}$$

The logistics formula previously discussed is adjusted to obtain a sigmoidal value (ξ) that is compatible with the XGBoost classifier and minimizes processing time.

$$\boldsymbol{\xi} = \boldsymbol{\delta^T} \cdot \boldsymbol{a} \tag{3}$$

Equation (3) depicts the fundamental linear modeling approach for any regression models used to create sigmoidal functions. Equation (4) expresses the fundamental sigmoidal activity with a bound $(-\infty, \infty)$.

$$\left(\mathbf{1}/[\mathbf{1} + \boldsymbol{e}^{-\boldsymbol{a}}]\right) = (\boldsymbol{e}^{\boldsymbol{a}}/[\mathbf{1} + \boldsymbol{e}^{\boldsymbol{a}}]) \tag{4}$$

Equations (5) and (6) estimate the correlation's likelihood function.

$$\mathbb{P}(\boldsymbol{x}) = [(\boldsymbol{\omega_0}) + (\boldsymbol{\omega_1 x_1}) + (\boldsymbol{\omega_2 x_2}) + \cdots + (\boldsymbol{\omega_k x_k})] \tag{5}$$

$$[\mathbb{P}(\boldsymbol{x})/[1 - \mathbb{P}(\boldsymbol{x})]] \Rightarrow [(\boldsymbol{\epsilon_0}) + (\boldsymbol{\epsilon_1 x_1}) + (\boldsymbol{\epsilon_2 x_2}) + \cdots + (\boldsymbol{\epsilon_k x_k})] \tag{6}$$

By considering both sides of the probabilities, the variation impact of independent variables on the reliant variable's value is estimated in Equation (7).

$$\mathbf{log}(\boldsymbol{x} \cdot \boldsymbol{\omega^t}) = \mathbf{log}[\mathbb{P}/(\mathbf{1} - \mathbb{P})] \tag{7}$$

After using the equation's inherent logarithmic feature (8),

$$\mathbf{log}\left[\mathbb{P}/(\mathbf{1} - \mathbb{P})\right] = \sum (\boldsymbol{x_j} \cdot \boldsymbol{\varepsilon_j}) \tag{8}$$

Hence the exponential formulae for LR is stated as

$$\mathbb{P} = \boldsymbol{e}^{(\boldsymbol{\varepsilon_j x_j})}/[\mathbf{1} + \boldsymbol{e}^{(\boldsymbol{\varepsilon_j x_j})}] \tag{9}$$

5 Inception V4

Inception V4.0 is recently published and incorporates features with significant modifications as well as upgrades for enhancing security and accessibility for administrators and clients. The preliminary batch of procedures prior to introducing the new layer has been adjusted in the 4^{th} edition of the model. Grid width and length (height) can be altered using specialised reducer components (blocks) in this version. This feature has been introduced to this version, while it was already included in prior versions. Inception V4's general structure is shown in the Figure 1.

5.1 Inception V4 with XRC

With the inclusion of modular components in the inception subsystems, Inception V4 reduces computational overhead. The inception module performs extensive computations using DL strategies that lead to efficient dimensionality reduction. Inception V4 considers factors like processing cost and generalization error (overfitting). It utilizes a variety of filters of varying sizes for parallelization. The Inception V4 contains a 1×1 auxiliary convolution operation for efficient computation and resilience. For classification, all the three datasets mentioned earlier are incorporated into the model using a dense layer (8×1). The approach is pre-computed using an efficient and robust cyber attack data features with a specified dimensionality feature ($224 \times 224 \times 3$) to reduce computational complexity. The suggested XRC model's general design is depicted in Figure 2. Table 4 delineates the entire procedure of XRC with Inception V4.

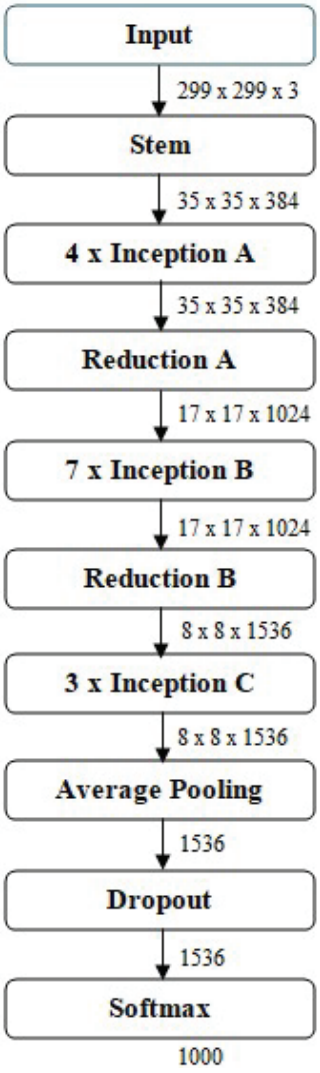

Figure 1 Inception V4 architecture.

To enhance the computational speed and precision of the suggested XRC method, the chained guideline and expectation-maximization properties are merged; the chained principle and expectation-maximization properties employed in this study are outlined in formula (10),

$$\boldsymbol{f}'(\boldsymbol{O}) = \boldsymbol{g}'(\boldsymbol{x}) \cdot \boldsymbol{f}'(\boldsymbol{g}(\boldsymbol{x})) \tag{10}$$

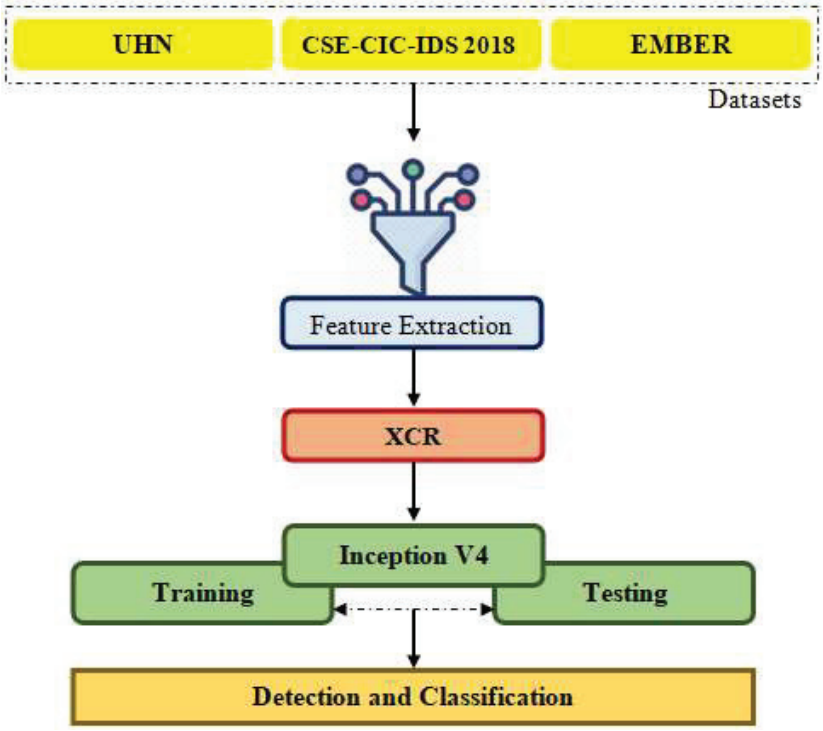

Figure 2 XCR Architecture integrated with inception V4.

Table 4 XCR algorithm to construct and process the tree

Step_1: Create decision tree (DT) based on computations of similarity score

$$\boldsymbol{\mu} = (\sum(\boldsymbol{\tau}))^{\mathbf{2}}/(\boldsymbol{\tau}^{\boldsymbol{n}} + \boldsymbol{\lambda})$$

Step_2: Suitable node is chosen based on the similarity score. The higher value indicates more homogeneity.

$$\boldsymbol{Selection} \Rightarrow (\boldsymbol{n_i}|\boldsymbol{\mu})$$

Step_3: Splitting of data using Gain information, $\boldsymbol{\eta}$

$$\boldsymbol{\eta} = (\boldsymbol{\mu_{LT}} + \boldsymbol{\mu_{RT}}) - \boldsymbol{\mu_r}$$

Step_4: Tree Pruning, $\boldsymbol{\psi}$

$$\boldsymbol{\psi} = (\boldsymbol{\eta} - \boldsymbol{\gamma}); \begin{cases} \boldsymbol{if}\ (\boldsymbol{\psi} > \mathbf{0}) \Rightarrow \boldsymbol{prune} \\ \boldsymbol{if}\ (\boldsymbol{\psi} \leq \mathbf{0}) \Rightarrow \boldsymbol{no\ prune} \end{cases}$$

Where $\boldsymbol{\gamma}$ is referred as complexity factor.

Step_5: Values for remaining nodes, μ_i

$$\boldsymbol{\mu_i} = [\sum(\boldsymbol{\tau})]/(\boldsymbol{\tau}^{\boldsymbol{n}} + \boldsymbol{\lambda})$$

After using the rule as mentioned above and simplifying, we get the equation as,

$$\boldsymbol{L} = (\mathbf{1} - \boldsymbol{L}(\boldsymbol{k})) \cdot \boldsymbol{L}(\boldsymbol{k}) \tag{11}$$

Expectation – maximization estimate for L is computed using Equations (12) and (13),

$$\widehat{\varepsilon}(\boldsymbol{r} : \boldsymbol{\theta}) = \left(\frac{\mathbf{1}}{\boldsymbol{n}}\right) \sum_{\boldsymbol{i-1}}^{\boldsymbol{n}} [\mathbf{ln}\boldsymbol{f}((\boldsymbol{\theta}|\boldsymbol{r_i}))] \tag{12}$$

$$\boldsymbol{L} = \sum \mathbf{log}(\mathbf{1} - \boldsymbol{L_i}(\boldsymbol{k_i})) + \sum \mathbf{log}(\boldsymbol{L}(\boldsymbol{k_i})) \tag{13}$$

The entropy evaluates of cyber attack via XRC is computed with Equation (14):

$$\boldsymbol{S} = -\sum (\mathbf{log}\,\boldsymbol{\alpha}^{\boldsymbol{i}} \times \boldsymbol{\alpha}^{\boldsymbol{i}}) \tag{14}$$

Equation (15) is used to compute attack events using the XRC model, which has been enhanced to include cyber attack capabilities.

$$\boldsymbol{\varphi} = \boldsymbol{\varphi} - \left[\frac{\boldsymbol{W}_{\boldsymbol{q}(\leftrightarrow)}^{\boldsymbol{adjusted}} \times \boldsymbol{\rho}}{\sqrt{\boldsymbol{\sigma} + \boldsymbol{T}_{\boldsymbol{q}(\leftrightarrow)}^{\boldsymbol{adjusted}}}}\right] \tag{15}$$

Where,

φ and $\leftrightarrow$ are referred as bias measures and weight;
Learning rate is indicated as 'ρ';
Gradient function is denoted as 'W';
Cyber event features are outlined as 'T' and
Scaling factor of crucial element (false alarm rate) is represented as 'σ'.

The entropy assessment with refinement is used to estimate the information losses during cyber events. Finally, Equation (16) expresses the XRC model for predicting the likelihood of a cyber attack.

$$\boldsymbol{C} = \{\mathbb{P}(\boldsymbol{C_1}), \mathbb{P}(\boldsymbol{C_2}) \cdots \mathbb{P}(\boldsymbol{C_n})\} \tag{16}$$

Many web-based intrusion detection tools have relied on Inception V4.0 as a valid reference to the training dataset. In addition to categorical and continuous characteristics, the data set used to train this classifier was labelled. This study is subjected to four different forms of attacks, namely, Distributed Denial of Service (DDoS), Phishing, Cross-site Scripting (CS), Internet of Things (IoT) Attacks.

DDoS: This type of cyber attack aims to render a host, operation, or infrastructure inaccessible by sending an excessive number of queries to the specific target source.

Phishing: These assaults are a blend of social manipulation and technological know-how. Using emails that seem to be from reliable sources, scammers might gather information from recipients or persuade them to do the desired action.

CS: Websites are infected by hackers who install destructive JavaScript into their databases. By visiting such a site, infected nodes receive the infected script included in the Html tags and activate it.

IoT Attack: IoT attacks occur when malicious actors attempt to breach the cybersecurity of an IoT network or equipment. IoT nodes that have been hacked might be used by criminals to extract private information, enlist a botnet, or takeover a network.

6 Outcomes and Discussion

Evaluation of the performance of the proposed XRC model integrated with inception V4 was discussed in this section. The performance measures that will be analysed are specified. They are precision, F1-Score, accuracy, Complete Prediction Measure (CPM), ROC (Receiver Operating Characteristics Curve), RMSE (Root Mean Square Error), and error rate, all taken into account while analysing cyberattacks classification. The estimated TP (True Positive), FP (False Positive), TN (True Negative), and FN (False Negative) values are used to assess the given parameters.

Accuracy (A): An overall amount of predictions was measured as the proportion of correct estimates, which is equated as,

$$\boldsymbol{A} = \frac{[\boldsymbol{TP} + \boldsymbol{TN}]}{[\boldsymbol{TP} + \boldsymbol{FP} + \boldsymbol{TN} + \boldsymbol{FN}]} \tag{17}$$

CPM: It is stated as the percentage of the overall predicted estimate that is correct and expressed as,

$$\boldsymbol{CPM} = \frac{[\boldsymbol{TP}]}{[\boldsymbol{FN} + \boldsymbol{TP}]} \tag{18}$$

Precision (P): The actual positive to overall expected ratio is shown and can be expressed as,

$$\boldsymbol{P} = \frac{[\boldsymbol{TP}]}{[\boldsymbol{TP} + \boldsymbol{FP}]} \tag{19}$$

F1-Score: It means the proportion among the overall mean of retention and the precision. In Equation (24), the F1-Score is stated as

$$\boldsymbol{F1}_{\boldsymbol{Score}} = \frac{[\boldsymbol{P} \times \boldsymbol{CPM}]}{[\boldsymbol{P} + \boldsymbol{CPM}]} \tag{20}$$

Specificity (ζ): It compares the positive classifications against the negative classifications and calculates the difference. Equation (21) defines it accordingly,

$$\zeta = \frac{[\boldsymbol{TN}]}{[\boldsymbol{FP} + \boldsymbol{TN}]} \tag{21}$$

ROC: It gives the categorization effectiveness or benchmark for the categorization of cyber attack events, which is depicted as,

$$\boldsymbol{ROC} = \frac{[\boldsymbol{TP}]}{[\boldsymbol{FN} + \boldsymbol{TP}]} \tag{22}$$

***RMSE*:** It demonstrates the variation in squared ratios between the estimated and absolute values using regression learner which is depicted as,

$$\boldsymbol{RMSE} = \sqrt{\sum_{i} (\boldsymbol{Estimated} - \boldsymbol{Absoute})} \tag{23}$$

Table 5 represent the allocated samples counts of different kinds of concerning attacks in all three datasets for the training and testing purpose.

6.1 Discussions on Evaluation and Performance

In this section, before comparing our method to the existing outcomes, we first present the outcomes with adjustable baseline weights and management of overfitting. The suggested XRC is compared to various existing techniques in terms of performance. Various approaches like Inception V3, convolution with eight layers, and SVM are compared to the recommended XRC, which is trained and tested through Inception V4. The result of RMSE for the

Table 5 Sample count of three dataset for four attacks

			Datasets			
			UHN	CSE-CIC-IDS2018	EMBER	Normal
Training	**Type of Attacks**	**DDoS**	386122	278456	354671	312034
		Phishing	218345	189379	243011	178934
		CS	88176	122093	78517	118342
		IoT attack	312056	234613	189372	232178
			Datasets			
			UHN	CSE-CIC-IDS2018	EMBER	Normal
Testing	**Type of Attacks**	**DDoS**	284321	240132	276712	217684
		Phishing	314175	165489	231046	167032
		CS	87341	127652	68210	102451
		IoT attack	212723	210976	167320	209761

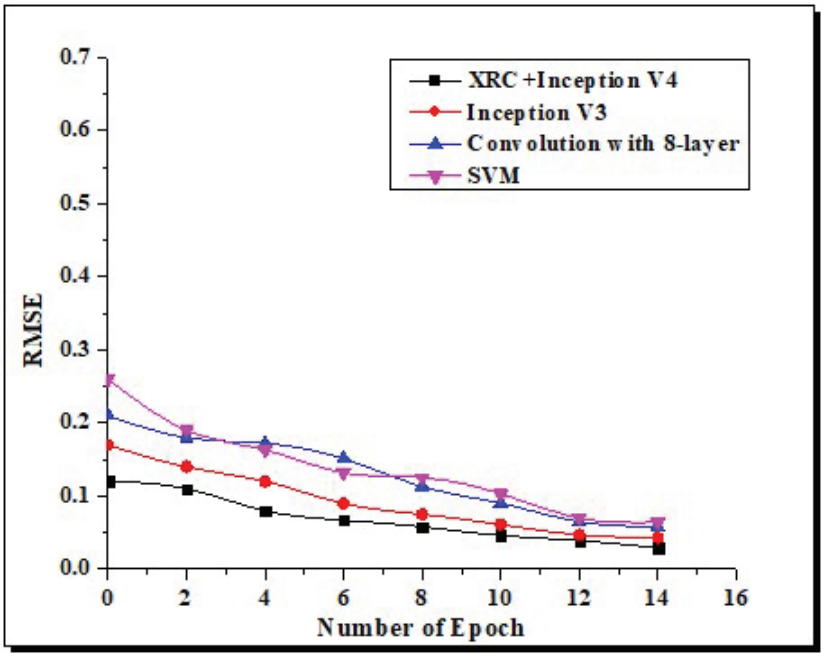

Figure 3 Error estimation during training phase.

concerned technique is shown in the figure for several epochs. Table 5 depicts the performance of proposed system through various metrics and it's been proved that the overall performance of proposed methodology is found to be better than the existing systems.

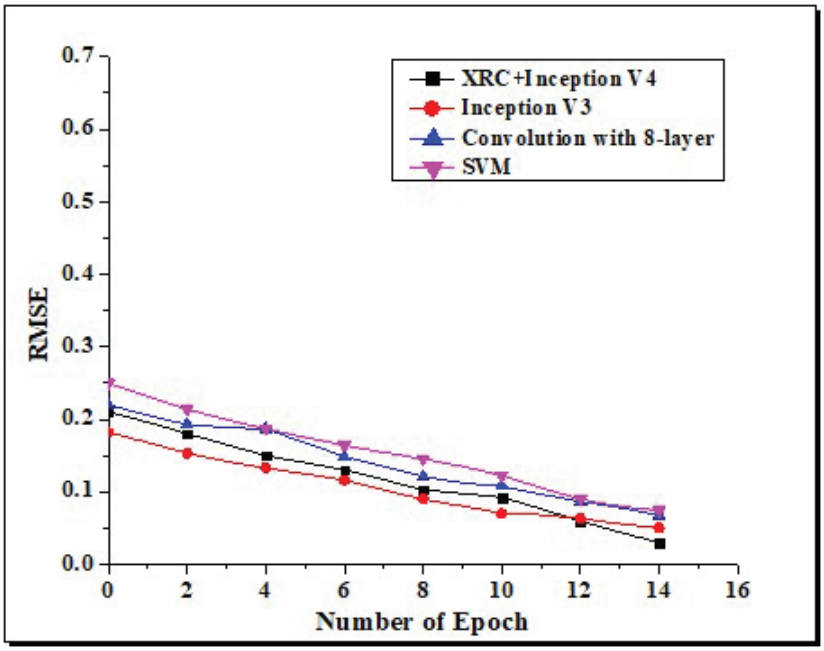

Figure 4 Estimation of validation error.

The classifier with the lowest RMSE delivers the best performance in detecting cyber attacks. According to a comparative study during the training phase, the suggested XRC has recorded the lowest RMSE compared to other approaches which is evident through the results depicted in Figure 3. Figure 4 shows a comparison of RMSE estimations for validation errors.

Comparing the existing systems and suggested XRC integrated with Inception V4, the RMSE estimate was much lower for the proposed method. The recommended XRC's RMSE value is much lower than the 0.02 compared to existing approaches. This ensures that the categorization of cyberattacks is accurate and error-free. Figure 5 shows the ROC prediction curve for the suggested XRC integrated with Inception V4.

The suggested XRC has recorded a ROC estimate of almost 0.1, making it a good prospect for regression modelling. The ROC is used to calculate the threshold levels for determining the kind of cyberattacks that occurred. There is no doubt that the suggested classifier can accurately categorize vulnerabilities with a low error rate. Figure 6 compares the error rate recorded during testing.

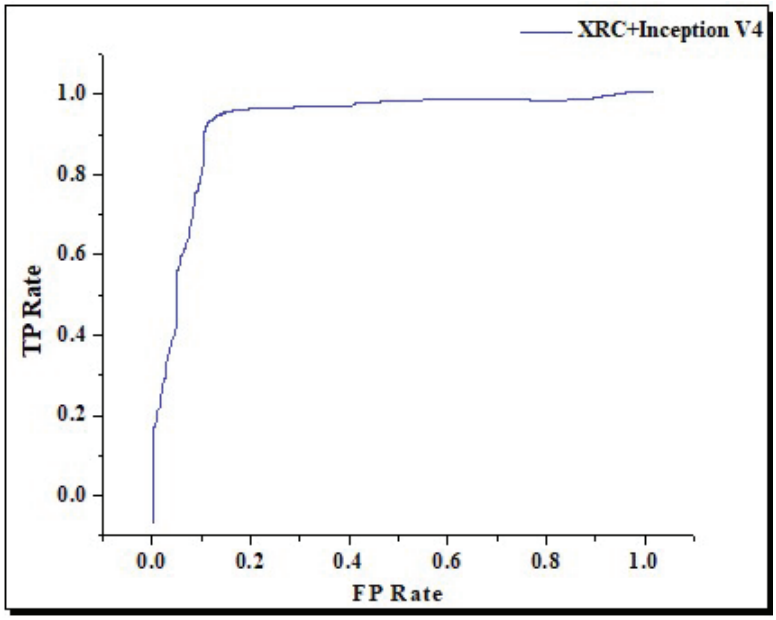

Figure 5 ROC curve for XRC + inception V4.

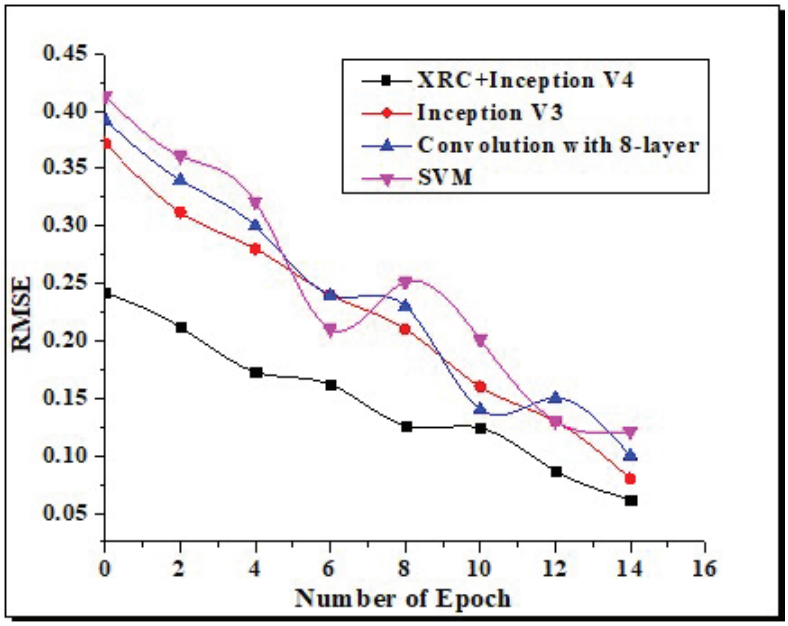

Figure 6 Analysis of test errors.

Table 6 Training and testing outcome of three dataset on XRC with inception V4

Training	Overall Detection Rate	Overall False-alarm Rate	Testing	Overall Detection Rate	Overall False-alarm Rate
	98.16%	2.316%		99.13%	2.136%

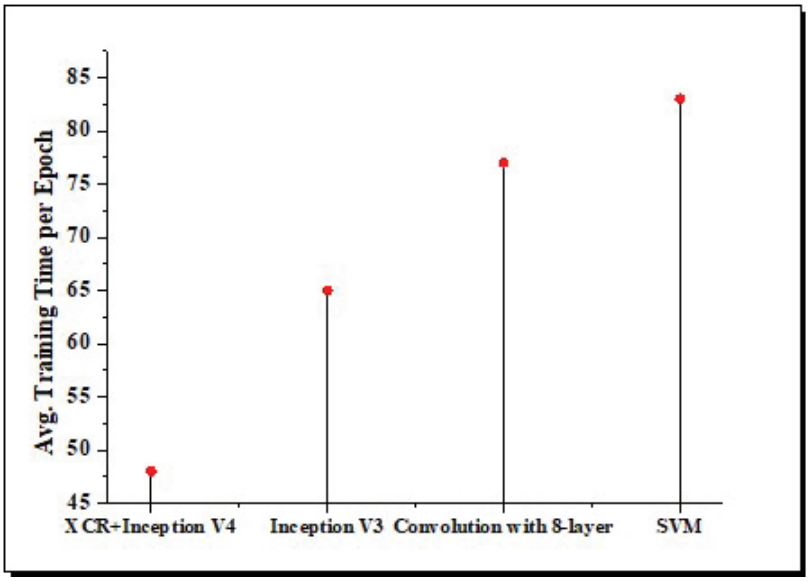

Figure 7 Comparison of avg. training time per epoch.

The suggested XRC has a low error rate during testing, just as in the training stage. The suggested XRC has recorded an error rate of less than 0.05, making it much less error-prone than existing approaches. This indicates that the suggested XRC has a lower error rate than the existing classification model.

Comparatively, the average training time per epoch is exhibited to show that the suggested XRC method takes less time to train the dataset than conventional methods. With Inception V4, the suggested XRC can be trained in about 48 seconds. The suggested XRC was also compared to existing techniques in terms of testing time. Figure 8 illustrates the average testing time per epoch for the recommended and the existing approaches.

Table 6 represents the overall resultant of both training and testing of three concerned dataset performed on XRC with Inception V4. Figure 7 depicts the

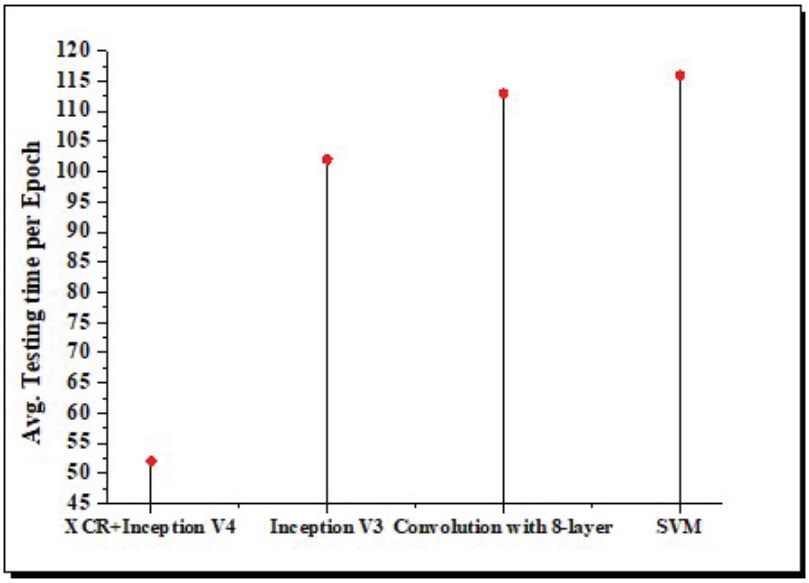

Figure 8 Average testing time per epoch.

amount of time it takes to process detection and classification of attack data during training.

To compare, a 52-second test run time for the new XRC method is about half the duration of previous methods. In addition, according to the study, the suggested XRC has recorded a low error rate for both training and testing.

7 Conclusion

Using an integrated classifier based on XGBoost, this work aims to reduce the detection error rate and computation time in training and testing the cyber attack datasets. XGBoost regression and Inception V4 classifiers were combined into the suggested architecture. This study observed that the suggested XRC is a very accurate and fast classifier of cyber events. Tests reveal that the proposed method has low error rates (of less than 0.05) and has recorded a ROC estimate of almost 0.1.In addition, the overall detection rate for both training and testing is 98.16% and 99.13%, respectively. The learning phase run time of the algorithm is quicker than that of other approaches. This analysis shows that our approach has the fastest execution time, which is a critical consideration in real-world applications. Computational complexity is

compatible with this conclusion. This study can be integrated with a security analysis system geared toward industrial use in the future.

References

[1] Z. Wang, L. Chen, S. Song, P. X. Cong, and Q. Ruan, "Automatic cyber security risk assessment based on fuzzy fractional ordinary differential equations," Alexandria Engineering Journal, vol. 59, no. 4, pp. 2725–2731, 2020.

[2] Van Staalduinen M. A, Khan F, Gadag V and Reniers G, "Functional quantitative security risk analysis (QSRA) to assist in protecting critical process infrastructure", Reliability Engineering & System Safety, vol. 157, pp. 23–34, 2017.

[3] A. Tantawy, S. Abdelwahed, A. Erradi, and K. Shaban, "Model-based risk assessment for cyber physical systems security," Computers & Security, vol. 96, p. 101864, 2020.

[4] C. Schmitz and S. Pape, "LiSRA: Lightweight Security Risk Assessment for decision support in information security," Computers & Security, vol. 90, pp. 101656, 2020.

[5] Venkatachary S. K, Prasad J and Samikannu R, "Cybersecurity and cyber terrorism-in energy sector–a review", Journal of Cyber Security Technology, vol. 2, no. 3, pp. 111–130, 2018.

[6] Kumar V. S, Prasad J and Samikannu R, "A critical review of cyber security and cyber terrorism–threats to critical infrastructure in the energy sector", International Journal of Critical Infrastructures, vol. 14, no. 2, pp. 101–119, 2018.

[7] Venkatachary S. K, Prasad J and Samikannu R, "Economic impacts of cyber security in energy sector: a review", International Journal of Energy Economics and Policy, vol. 7, no. 5, pp. 250–262, 2017.

[8] Venkatachary S. K, Prasad J and Samikannu R, Alagappan A and Andrews L. J. B, "Cybersecurity infrastructure challenges in IoT based virtual power plants", Journal of Statistics and Management Systems, vol. 23, no. 2, pp. 263–276, 2020.

[9] Benaroch M, "Real options models for proactive uncertainty-reducing mitigations and applications in cybersecurity investment decision making", Information Systems Research, vol. 29, no. 2, pp. 315–340, 2018.

[10] A. Nhlabatsi et al., "Threat-Specific Security Risk Evaluation in the Cloud," in IEEE Transactions on Cloud Computing, vol. 9, no. 2, pp. 793–806, 2021.
[11] Khidzir N. Z, Daud K. A. M, Ismail A. R, Ghani M. S. A. A and Ibrahim M. A. H, "Information Security Requirement: The Relationship Between Cybersecurity Risk Confidentiality, Integrity and Availability in Digital Social Media", Regional Conference on Science, Technology and Social Sciences (RCSTSS), pp. 229–237, 2018.
[12] Kusyk J, Uyar M. U and Sahin C. S, "Survey on evolutionary computation methods for cybersecurity of mobile ad hoc networks", Evolutionary Intelligence, vol. 10, no. 3, pp. 95–117, 2018.
[13] Sampathkumar, A., and Vivekanandan, P, Gene Selection Using Parallel Lion Optimization Method in Microarray Data for Cancer Classification. Journal of Medical Imaging and Health Informatics, vol. 9, no. 6, pp. 1294–1300, 2019.
[14] Ashibani Y and Mahmoud Q. H, "Cyber physical systems security: Analysis, challenges and solutions", Computers & Security, vol. 68, pp. 81–97, 2017.
[15] Sampathkumar, A., Maheswar, P& Hashvardhan, "Majority Voting based Hybrid Ensemble Classification Approach for Predicting Parking Availability in Smart City based on IoT", 11th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1–8, 2020.
[16] Abdo H, Kaouk M, Flaus J. M and Masse F, "A safety/security risk analysis approach of Industrial Control Systems: A cyber bowtie–combining new version of attack tree with bowtie analysis", Computers & Security, vol. 72, pp. 175–195, 2018.
[17] Urbina D. I, Giraldo J. A, Cardenas A. A, Tippenhauer N. O, Valente J, Faisal M and Sandberg H, "Limiting the impact of stealthy attacks on industrial control systems", Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, pp. 1092–1105, 2016.
[18] A. Gupta, A. Anpalagan, G. H. S. Carvalho, A. S. Khwaja, L. Guan, and I. Woungang, "RETRACTED: Prevailing and emerging cyber threats and security practices in IoT-Enabled smart grids: A survey," Journal of Network and Computer Applications, vol. 132, pp. 118–148, 2019.
[19] Januário F, Cardoso A and Gil P, "A distributed multi-agent framework for resilience enhancement in cyber-physical systems", IEEE Access, vol. 7, pp. 31342–31357, 2019.

[20] Durand L, "Cyber security: a risky business", 2018. https://studentthese s.universiteitleiden.nl/access/item%3A2666281/view
[21] Wu Z, Albalawi F, Zhang J, Zhang Z, Durand H and Christofides P. D, "Detecting and handling cyber-attacks in model predictive control of chemical processes", Mathematics, vol. 6, no. 10, 2018.
[22] Sándor H, Genge B, Szántó Z, Márton L and Haller P, "Cyber attack detection and mitigation: Software defined survivable industrial control systems", International Journal of Critical Infrastructure Protection, vol. 25, pp. 152–168, 2019.
[23] Paoletti N, Jiang Z, Islam M. A, Abbas H, Mangharam R, Lin S and Smolka S. A, "Synthesizing stealthy reprogramming attacks on cardiac devices", Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems, pp. 13–22, 2019.
[24] Liu L, De Vel O, Han Q. L, Zhang J and Xiang Y, "Detecting and preventing cyber insider threats: A survey", IEEE Communications Surveys & Tutorials, vol. 20, no. 2, pp. 1397–1417, 2018.
[25] Dataset of UHN, EMBER: https://csr.lanl.gov/data/2017/
[26] Dataset of CSE-CIC-IDS 2018, https://www.kaggle.com/solarmainfram e/ids-intrusion-csv
[27] L. Lorenzi, "Analytical Methods for Kolmogorov Equations," Oct. 2016.
[28] J. Milosevic, H. Sandberg, and K. H. Johansson, "Estimating the Impact of Cyber-Attack Strategies for Stochastic Networked Control Systems," IEEE Transactions on Control of Network Systems, vol. 7, no. 2, pp. 747–757, Jun. 2020.
[29] R. Hoffman, "The General Cyber-Attack Life Cycle And Its Continuous-Time Markov Chain Model," Ekonomiczne Problemy Usług, vol. 131, pp. 121–130, 2018.
[30] H. Om and T. K. Sarkar, "Designing Intrusion Detection System for Web Documents Using Neural Network," Communications and Network, vol. 02, no. 01, pp. 54–61, 2010.
[31] M. E. Haque and T. M. Alkharobi, "Adaptive Hybrid Model for Network Intrusion Detection and Comparison among Machine Learning Algorithms," International Journal of Machine Learning and Computing, vol. 5, no. 1, pp. 17–23, Feb. 2015.
[32] G. R. Kumar, N. Mangathayaru, and G. Narsimha, "An approach for intrusion detection using fuzzy feature clustering," 2016 International Conference on Engineering & MIS (ICEMIS), Sep. 2016.

[33] C. Liu, J. Yang, and J. Wu, "Web intrusion detection system combined with feature analysis and SVM optimization," EURASIP Journal on Wireless Communications and Networking, vol. 2020, no. 1, Feb. 2020.
[34] S. S. Sivatha Sindhu, S. Geetha, and A. Kannan, "Decision tree based light weight intrusion detection using a wrapper approach," Expert Systems with Applications, vol. 39, no. 1, pp. 129–141, Jan. 2012.
[35] T. A. Deepak, "XGBoost Classification based Network Intrusion Detection System for Big Data using PySparkling Water," International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 1, pp. 377–382, Feb. 2020.
[36] A.-C. Enache and V. Sgârciu, "Enhanced Intrusion Detection System Based on Bat Algorithm-support Vector Machine," Proceedings of the 11th International Conference on Security and Cryptography, 2014.

Biographies

K. M. Karthick Raghunath, is an Associate Professor in the Computer Science and Engineering Department in MVJ College of Engineering, Bangalore, India. He has received his B. Tech., in Information Technology from Anna University in 2008 and M.E., in Pervasive Computing Technology from Anna University (BIT Campus) in 2011. In 2019, he completed his Ph.D. degree from Anna University, Chennai. With nearly a decade of experience in teaching, his areas of specialization include pervasive computing, Artificial Intelligence, IoT, Data Science, and WSN.

V. Vinoth Kumar is an Associate Professor at Department of Computer Science, JAIN (Deemed-to-be University), Bangalore, India. His current research interests include Wireless Networks, Internet of Things, machine learning and Big Data Applications. He is the author/co-author of papers in international journals and conferences including SCI indexed papers. He has published as over than 35 papers in IEEE Access, Springer, Elsevier, IGI Global, Emerald etc.. He is the Associate Editor of International Journal of e-Collaboration (IJeC), International Journal of Pervasive Computing and Communications (IJPCC) and Editorial member of various journals.

Muthukumaran Venkatesan is working as an Assistant Professor in the Department of Mathematics, REVA University Bangalore, India. He received the B.Sc. degree in Mathematics from the Thiruvalluvar University Serkkadu, Vellore, India, in 2009, and the M. Sc. degrees in Mathematics from the Thiruvalluvar University Serkkadu, Vellore, India, in 2012. The M. Phil. Mathematics from the Thiruvalluvar University Serkkadu, Vellore, India, in 2014 and Ph.D. degrees in Mathematics from the School of Advanced Sciences, Vellore Institute of Technology, Vellore in 2019. His current research interests include Fuzzy Algebra, Fuzzy Image Processing, Data Mining, and Cryptography.

Krishna Kant Singh is working as Professor, Faculty of Engineering & Technology, Jain (Deemed-to-be University), Bengaluru, India. He has wide teaching and research experience. Dr. Singh has acquired B.Tech, M.Tech, and Ph.D. (IIT Roorkee) in the area of image processing and Machine Learning. He has authored more than 90 research papers in Scopus and SCIE indexed journals of repute. He has also authored 25 technical books. He is also an associate editor of IEEE ACCESS (SCIE Indexed) and Guest Editor of Microprocessors and Microsystems, Wireless Personal Communications, Complex & Intelligent Systems. He is also member of Editorial board of Applied Computing and Geoscience (Elsevier). Dr. Singh is an active researcher in the field of Machine Learning, Cognitive Computing, 6G and beyond networks.

T. R. Mahesh has received Bachelor of Engineering, Master of Technology and Doctorate of Philosophy in Computer Science and Engineering and he is carrying out research in the area of Data mining, machine learning, artificial intelligence and web mining. He has more than 20 years of experience in academics and has served at various levels. He has published various papers in National and International reputed journals. Currently he is serving

as Associate Professor and Program Head in the Department of Computer Science and Engineering at Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Bengaluru.

Akansha Singh is working as Associate Professor in School of Computer Science and Engineering, Bennett University, Greater Noida, India. She is B.Tech, M.Tech and PhD in Computer Science. She received her PhD from IIT Roorkee in the area of image processing and machine learning. Dr. Singh has to her credit more than 70 research papers, 20 books and numerous conference papers. She has been the editor for books on emerging topics with publishers like Elsevier, Taylor and Francis, Wiley etc. Dr. Singh has served as reviewer and technical committee member for multiple conferences and journals of High Repute. She is also the Associate Editor for IEEE Access and Open Computer Science journal. Dr. Singh has also undertaken government funded project as Principal Investigator. Her research areas include image processing, remote sensing, IoT and machine learning.