Mesopotamian journal of Big Data
Vol. **(2024)**, 2024, **pp**. 230–240
DOI: https://doi.org/10.58496/MJBD/2024/017 ISSN: 2958-6453
https://mesopotamian.press/journals/index.php/BigData

Research Article

Anomaly-Based IDS (Intrusion Detection System) for Cyber-Physical Systems

Ahmad Muter Awaad [1,*], Khattab M Ali Alheeti [2], Abdul Kream A.H. Najem [2]

[1] *Directorate of Education in Al-Anbar, Iraq.*
[2] *Computer Science and Information Technology in University of Anbar, Iraq.*

ARTICLE INFO

Article History
Received 15 Oct 2024
Accepted 14 Nov 2024
Published 06 Dec 2024

Keywords
Cyber-physical systems (CPS)
Anomaly-based intrusion detection
Machine learning
Critical infrastructure security
Cyber threats
Classification techniques
Intrusion detection system (IDS)

ABSTRACT

Cyber-physical systems (CPS) are critical infrastructures that integrate physical processes with computational components. The security of CPS is paramount, as any breach can lead to severe consequences. Anomaly-based intrusion detection systems (IDS) have emerged as a promising approach to safeguard CPS against cyber threats. This paper presents an anomaly-based IDS designed specifically for CPS, leveraging machine learning techniques to establish a baseline of normal system behaviour and promptly detect deviations indicative of malicious activities. The proposed system incorporates multiple classification techniques, including KNeighbors, RandomForest, XGB, DecisionTree, SGD, SVM, LGBM, AdaBoost, Bagging, and MLP Classifier, to enhance detection accuracy and robustness. Key components of the IDS, such as data collection, feature extraction, anomaly detection, and alert generation, are thoroughly outlined. The system's performance is evaluated, highlighting its effectiveness in accurately identifying intrusions while maintaining low false positive rates. The proposed anomaly-based IDS aims to provide a robust and reliable solution for enhancing the security of CPS and protecting critical infrastructure from cyber threats.

1. INTRODUCTION

A "cyber physical system" (CPS) is a networked collection of sensors, actuators, control and networking components, and other elements that work together to perform a key function, regardless of their location or kind. Communication and information systems (CPSs) include technologies like smart grids, unmanned aircraft systems, ubiquitous healthcare systems, and first responder situational awareness systems. These systems are characterised by having several control loops, demanding precise timing, reliably predicting network traffic, using older components, and even include wireless network segments. Cyber-physical systems integrate many elements from both the cyber (network components and commodity servers) and physical (sensors and actuators) realms.

During the course of a CPS's operation, the assault model incorporates both immediate and delayed strikes. An adversary who isn't careful enough to secure the network might quickly disrupt the relevant processes, leading to a catastrophicoutcome. However, a skilled attacker may be careful not to disrupt normal system operations while spreading and building up a dispersed attack that began at one place. This is the kind of attack that Stuxnet used, as stated by Keizer (2010) and Stuxnet (2013). This is the reason why the most significant problem in the design of a CPS intrusion detection system (IDS) is the speed of detection, also known as detection latency. Utilizing their one-of-a-kind characteristics and identifying assaults that are not previously known are the primary focuses of the creation of CPS intrusion detection systems.

The architecture of industrial control systems (ICSs), which were formerly isolated networks, has transitioned to a network that is now known as Networked Control Systems (NCS). This occurred due to many causes, such as technological advancements, the integration of Internet of Things (IoT) in the industrial sector, and the emergence of various applications associated with industry 4.0. Due to these conditions, a shift has taken place. SCADA systems, short for supervision, control,

*Corresponding author. Email: it@gmail.com

and data acquisition, are used for the purpose of overseeing and managing duties such as monitoring and control across large geographical regions. Communication methods are used to exert control over these actions. This technical advancement has led to significant improvements in performance. However, it has also led to a greater variety of possible threats that these systems may be exposed to. As a result, these systems have shown substantial improvement in performance. The main point is that oppositional groups or dissatisfied workers are being seen as potential sources of hazards [1]. Moreover, when considering the fact that these systems are frequently employed to monitor and regulate critical infrastructure, the matter of their safety becomes even more crucial as any malfunctions could potentially endanger the general public. The increasing interest in research pertaining to cyber-physical security is to be expected, given that it poses a considerable challenge in the management of existing industrial control systems (ICS). This is because this is a factor that is considered. Regarding information technology (IT), three essential concepts must be given priority to ensure the safety and security of associated data and resources [2]. These principles are commonly referred to as the CIA, which stands for confidentiality, integrity, and availability. In the industrial sector, AIC [3] presents an inverted perspective about security. This is in contrast to the commonly utilised method in the area of information technology, which follows the CIA paradigm in terms of the prioritisation of the listed criteria.

Furthermore, it is essential to safeguard data in operational technology (OT) in order to ensure the continued availability of plants and resources. This is because the absence of some resources might result in significant harm to both people and the environment. The cyber-attacks that involve many sorts of vital infrastructures serve as the impetus for our efforts. The assault known as Stuxnet is one of them [4]. The Stuxnet virus was used to alter Programmable Logic Controllers (PLCs) that oversee the regulation of centrifuges at a nuclear power plant. This hindered the identification of the subsequent malfunctions. The program's intricacy, apparent upon examination, originally captured the interest of specialists. The virus most probably began from inside the plant itself, perhaps delivered by an infected USB key accidentally carried by a person. By using four Windows vulnerabilities, the malicious malware obtained network access and propagated itself in order to locate PLC control software. Subsequently, it altered its code to inflict harm on the system without implementing any precautionary measures, finally notifying the operators of the abnormality. Important attack events are given in Table 1, each of which has caused significant damage to global industry. Therefore, there is a growing interest in cyber-attacks on CPSs.[5]

TABLE I. TYPICAL CYBER ATTACK EVENTS FROM YEARS 2010 TO 2021

Year	Country/institution	Details
2010	Iran	Stuxnet attack destroying core controllers of industries
2015	Ukraine	Black Energy attack on power grid, leading to massive power 0 utage
2017	Russia, Ukraine, India, China	WannaCry attack aiming to encrypt data and demand ransom payments
2020	Brno University Hospital, CzechA cyber-attack that shut down Republic IT network of a Czech hospital	
2020	CJS Dept. Health & Human Services	Unspecified attack on servers
2021	Colonial Pipeline, US	A ransomware attack on a US fuel pipeline, leading to shutdown of a critical fuel network

To ensure the safety of a computer system, it is necessary to have an architecture that is composed of various kinds of hardware devices and software programs. This design should be able to create a barrier that is capable of defending the system from the many kinds of assaults to which it is vulnerable. As a result, there is a need for a technology that is able to monitor these kinds of systems and, as a result, uncover any potential security breaches. This particular piece of technology is referred to as the Intrusion Detection System (IDS). Intrusion detection is the act of monitoring and assessing events happening within a computer system or network to identify any efforts to undermine the confidentiality, integrity, or availability of data or the network [6]. To integrate the data from the regulated process, the IDS technology utilised in SCADA systems must be strengthened [7].

Utilizing a supervised Machine Learning (ML) technique, the purpose of this research is to suggest the construction of a defensive solution for industrial control systems (ICS). In order to identify any deviations from the typical behaviour of the system, a classifier is subsequently acquired.

2. RELATED WORKS

While many research focus on intrusion detection methods in traditional networks and communication systems, very few examine CPSs and evaluate ML methods both online and offline. One reason why traditional offline ML IDSs cannot run on these systems is because they can't monitor and understand data from all those sources at once [8]. Additionally, there is a dearth of research that measures the IDS's latency or resource usage when using CPS datasets.

Jun Inoue et al. [9] compared the detection abilities of two unsupervised ML algorithms: a one-class SVM and a Deep Neural Network (DNN) with an LSTM layer followed by several feed-forward layers. The first one's like a simplified one-class support vector machine (SVM), while the second one uses probabilistic outlier identification to identify data points with low probability as outliers. The DNN got a f1-score of 80.3% on the SWaT dataset, whereas the one-class SVM got a little lower result of 79.6% for the same measure. The scientists have also determined that the two approaches have similar shortcomings when it comes to finding subtle shifts in sensor readings and unusual actuator activity.

Using a 1D-CNN, Moshe Kravchik et al. [10] were able to detect outliers in the SWaT dataset. The training dataset's prediction error is used to estimate a threshold. By using a z-score function and such a threshold, the authors successfully differentiated between assaults and typical activities. In order to gain a decent notion of the optimal hyperparameters for the local region, a grid search was used. It was in this manner that an F1-score of 87.1% was attained. The hyper parameters of an Autoencoder network were fine-tuned in [11] using a mix of grid search and evolutionary algorithms. The author utilised the previous technique for anomaly detection. When compared to his previous attempts, the author's current F1-score of 87.3% is much higher. Using a model based on Long-Short Term Memory (LSTM), Giulio Zizzo et al. [12] were able to attain an F1-score of 81.7%. For optimal results, determine the window's duration and error threshold using grid search.

The anomalous detection rate of several kinds of neural networks for spotting assaults on a CPS was compared by Dmitri Shalyga et al. in [13]. Predictions of future sensor readings were made using data from a fixed-duration window in the past. Any deviation from the 99th percentile in the anticipated error between the predicted and actual values is considered an anomaly. In addition, a number of methods were used, including weighted powered error, mean p-powered error, and exponentially weighted smoothing, to enhance the credibility of the anomalous threshold. Lastly, a fragmented time window approach was utilised to address the imbalanced dataset. As a result, the window utilised for making predictions at a given moment may not always be the one that was just before it. In this particular configuration, an MLP model attained an F1-score of 81.2%.

An innovative method for anomaly detection using Generative Adversarial Networks (GAN) is presented by D. Li et al. in their paper [14]. One LSTM Recurrent Neural Network (LSTMRNN) is trained to identify out-of-the-ordinary occurrences, while the other is taught to generate synthetic normal baselines; this training strategy is used in this study. They reach an F1-score of 77% on the SWaT testbed after including these two models as adversaries in an iterative feedback-loop training technique.

M. Elnour et al. achieved the highest results in the literature for this dataset with an F1-score of 88.2% in their publication [15]. By comparing the original data with its modified representations using Principal Component Analysis (PCA), this isolation forest based anomaly detection method may distinguish abnormal observations from typical ones.

Three steps are outlined in the suggested approach for data analysis in categorization jobs. The dataset is split into training (80%) and testing (20%) sets during the first round of data pre-processing, which also fixes missing values. Second, several classification techniques are used, such as AdaBoost, LightGBM, bagging, multi-layer perceptrons, stochastic gradient descent, KNN, decision trees, SVM, random forests, and XGBoost. In conclusion, assessment uses metrics like accuracy, precision, recall, and F1-score to assess the performance of the mathematical framework.

3.PROPOSED METHODOLOGY

In this figure it's explain the proposed methodology for 10 classifiers model how it work

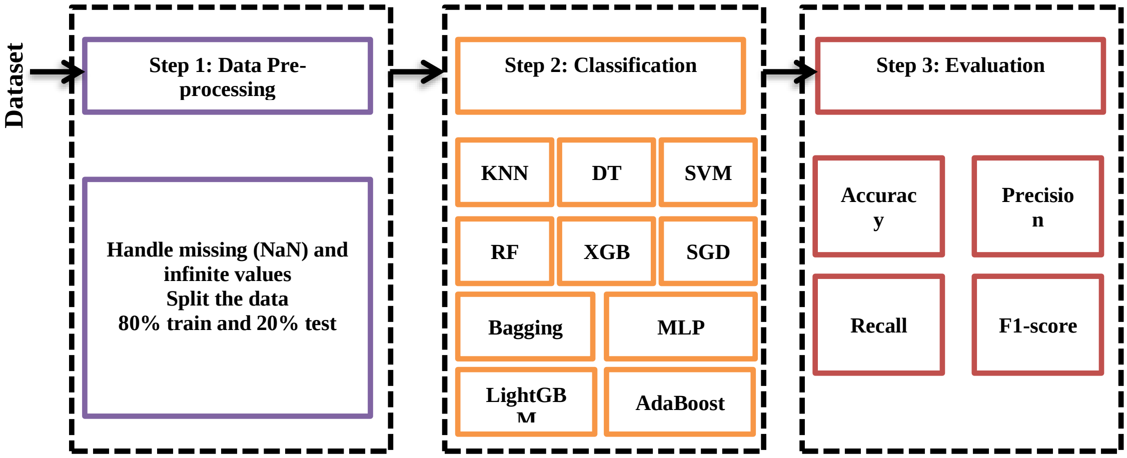

Fig. 1. Life Cycle Of Model

The diagram outlines a three-step process for data analysis, specifically focusing on classification tasks. Here’s a breakdown of each step:

Step 1: Data Pre-processing

- Handle Missing Values: Address any missing (NaN) or infinite values in the dataset to ensure data quality.
- Data Splitting: Divide the dataset into training and testing sets, typically using 80% of the data for training and 20% for testing.

Step 2: Classification

This step involves applying various classification algorithms to the pre-processed data. The algorithms listed include:

- KNN (K-Nearest Neighbors)
- DT (Decision Tree)
- SVM (Support Vector Machine)
- RF (Random Forest)
- XGB (XGBoost)
- SGD (Stochastic Gradient Descent)
- Bagging
- MLP (Multi-Layer Perceptron)
- LightGBM
- AdaBoost

Step 3: Evaluation

After classification, the model's performance is evaluated using several metrics:

- **Accuracy:** The proportion of true results among the total number of cases examined.
- **Precision:** The ratio of correctly predicted positive observations to the total predicted positives.
- **Recall:** The ratio of correctly predicted positive observations to all actual positives.
- **F1-score:** The harmonic mean of precision and recall, providing a balance between the two.

This structured approach ensures a systematic method for handling data, applying classification techniques, and evaluating model performance**.**

3.1 Attacker Model

Evidence from previous massive assaults on IoT devices, such as the Mirai botnet, suggests that infecting the maximum number of accessible devices is the objective of the attacker. It is usual practice to gather data from all accessible devices during the reconnaissance phase of cyber kill chains. Recent attacks against CPSs have included these, as shown by assessments done by [16]. Due to the absence of security controls on operational technology (OT) platforms used for CPS, it is also assumed that every device has a vulnerability that an attacker may exploit to access or modify it.

We anticipate that a lone intruder will be able to access the whole system network via a single point of entry. Specifically, other systems could be open to remote assaults. At the same time, it's reasonable to suppose that the attacker intends to penetrate the target network from inside, as this would facilitate the attack's propagation.

Multiple simultaneous attacks from many attackers are theoretically feasible in a network setting. Nevertheless, we operate on the premise that individual attacks do not influence the actions of an individual attacker since they do not interact with each other. So that we don't miss the mark on generalisation, we always portray only one attacker.

We will now go over two examples of how our modelling framework may be put to use in the following paragraphs. We begin by using the local alert log files on a single host in order to get an understanding of the path that assaults take across that host. Because of this, we refer to this as the propagation and tracing of an intra-host attack. The second thing that we do is expand this notion to assaults that are directed at computer networks.

3.2 Intra-host Anomaly Detection

When a single host is taken into consideration, there are many sorts of logs that give information for anomaly detection. There is a possibility that application layer logs will include information on unusual occurrences pertaining to corresponding functionalities. In addition to providing information about the overall condition of the system, general system logs also make it possible to identify low-level abnormalities and assaults. Additionally, a recording or detection layer that records suspicious network activity might be included into every communication interface in the future. This configuration shown for two hosts is seen in Figure 1. Depending on the host, an attacker might use one of three possible attack dissemination paths.

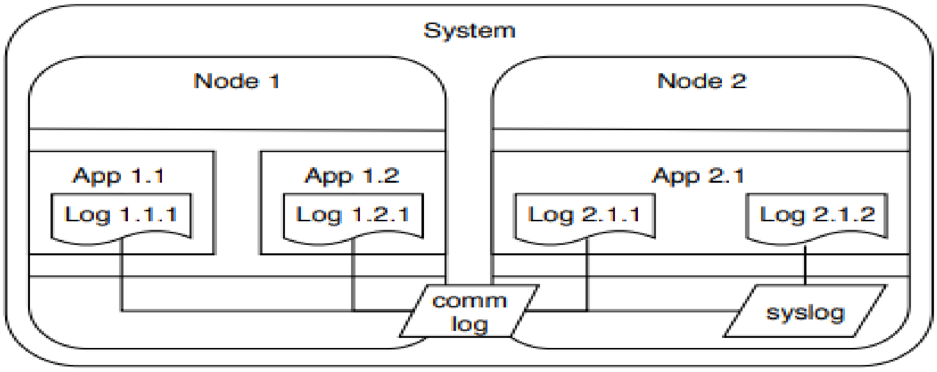

Fig. 2. Locating accessible alert log files and understanding their relationships

To begin, a remote attacker has the ability to change the behaviour of the program by exploiting a vulnerability in the application logic. This results in an anomaly being recorded in the application log. As a result of this modification, it is possible that further assaults may be launched against the underlying system. As a result, the attacker has the ability to exert influence on the state of the system and expand its privileges so that they may spread to other hosts by using the availability of communication interfaces. Therefore, the intrusion detection system (IDS) on the communication interface is responsible for identifying the unusual network traffic.

This, in turn, has an effect on the way the system processes the communication, which ultimately results in a system that has been influenced. Following that, the application logic of each and every program running on this system is open to further tampering.

Communication → (system) → application

Furthermore, the system's state might be directly altered by an attack on the hardware or other local assaults. A compromised system's communication layers allow an attacker to spread their malware to other applications or even other computers.

$$\text{System} \rightarrow \text{application} \mid \text{communication}$$

It is essential to keep in mind, with regard to the identification of anomalies, that it is possible for an attack to spread without causing the underlying system to enter an abnormal condition. Creating malicious network packets that adhere to the requirements but include malicious payloads is one method that may be used to accomplish this goal. Therefore, logs depicting the current status of the system can be less significant for detection.

On a more abstract level, we outlined these intra-host pathways; yet, they serve as the foundation for any assault that can be launched against any system. Research shows that real-life attackers often use these channels in conjunction with one another to launch complex multi-stage attacks. We presume that detections on adjacent regions of the model that happen consecutively indicate the assault's origin and destination in order to locate such attack chains. As a result of the stated attack propagation paths, the sequence of detections in the multiple linked alert log files may be able to reveal the sort of attack associated with each detection as cause-and-effect relationships are formed.

3.3 Inter-host Anomaly Detection

In contrast to the straightforward configuration shown in Figure 2, network systems that are used in the real world often consist of a much greater number of independent hosts that are organized in a variety of topologies. In light of this, it is necessary for us to additionally explore the features of attack propagation in bigger networks.

G = (V, E) is an architecture that we evaluate for so that larger networks might undergo an investigation of distributed anomaly detection system installations. This design is comprised of systems V and connections E. In an undirected graph, any system v that is a member of V is regarded as a node. However, in that specific graph, the edge ei,j = (vi, vj) represents the same link as the one between vi and vj.

Systems that are physically close to one another, linked to the same network, or utilised by the same people are all examples of links in real-world system modelling. By doing so, we may also mimic social engineering attack routes and assaults that traverse air gaps, such USB attacks.

Two events, Ai, t and Di, t are used to describe the state of the system vi at time t. Attacks on these events occur, while detections of them take place. Furthermore, the equation ai = (di, fi) may represent an anomaly detection system that can be implemented in any system in the network. The detection and false alarm rates shown by this specific detection system are indicated. An continuing attack on the system is likely to be detected, and the detection rate reflects that possibility.

$$d_i = P\left(D_{i,t+1} \,\middle|\, A_i, t\right) \tag{1}$$

The likelihood of an assault being detected while none is really happening is known as the false alarm rate. This implies that

$$f_i = P\left(D_{i,t} \,\middle|\, \backsim A_i, t\right) \tag{2}$$

By combining the accuracy and recall measures, we can construct a broad evaluation of the network's anomaly detection systems' performance. All of the systems in the network may contribute to this assessment by adding their detection categories.

$$Precision = \frac{t_p}{t_p + f_p} \tag{3}$$

$$recall = \frac{t_p}{t_p + f_n} \tag{4}$$

We refer to these measures as the node recall and precision since they show how well the nodes as a whole perform when it comes to detecting anomalies.

For the sake of describing the network's attack propagation, it is assumed that a compromised system would transmit assaults to neighbouring systems adj (si) with a probability of ii in two consecutive units of time.

$$i_i = P\left(A_{j,k} \,\middle|\, A_i k - p\right) \tag{5}$$

with p > 0, ∀sj ∈ adj(si)

$$P(A_{j,k}) = \sum_{s_i \in adj(s_j)} \sum_{t=0}^{k-1} i_i . P(A_{i,t}) \tag{6}$$

Through the development of this attack propagation model, it has been shown that the exposure of a system to other compromised systems enhances the risk of its own penetration. Not only does this exposure grow with time, Together with the system's degree on the graph, it likewise increases. An equation that may be used to determine the degree k of a node n in a graph is k = |adj (n)|. Consequently, it stands to reason that attacks would propagate more rapidly across networks with a higher connection level, or median degree of connectedness.

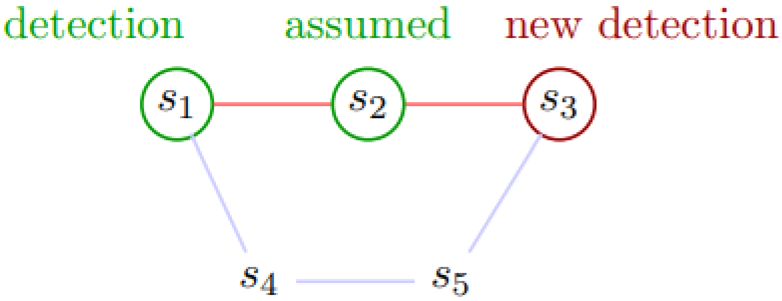

Fig. 3. Visualization of presumed identifications along the most direct routes

Take into consideration the route s1 ⟶ s2 ⟶ s3 that has been described as the shortest path for a fresh assault on node n3. Due to the fact that we only evaluate pathways that begin at nodes that have previously been attacked, s1 must be assaulted. If, on the other hand, s2 was assaulted, the route from s2 to s3 would be shorter since it is a subsequence of the initial path's sequence. Accordingly, the route would already come to a stop at the place of s2. As a result, we consider all of the intermittent nodes that are located along these lines to be presumed detections. A visual representation of our idea of assumed detections on shortest pathways is shown in Figure 3. Because we do not have any other information, it is also plausible that the attacker really used the lengthier route, which is s1 ⟶ s4 ⟶ s5 ⟶ s3. When we pick the needed effort for an attacker as a distance metric between nodes, however, it is at least more probable that the shorter route has been used. This is because the shorter way costs less effort to take. The results of such efforts may, for instance, be gleaned from SRAs. By distinguishing between assumed detections and other detections, we can assure that the estimated attack pathways will not be altered throughout later iterations of Algorithm 1.

3.4 Dataset Summary

Power System Attack Datasets - Mississippi State University and Oak Ridge National Laboratory - 4/15/2014 There are three datasets contained in this folder. They are made from one initial dataset consisting of fifteen sets with 37 power system event scenarios in each. The multiclass datasets are in ARFF format for easy use with Weka and the others are in CSV format alsocompatible with Weka.

Here’s how we can create a table for the dataset used, a comparison table for all the machine learning techniques, and plot the metrics for each classifier using the given dataset and classification results.

First, we'll start with creating a table that summarizes the dataset and a comparison table of all ML techniques:

TABLE II. DATASET DETAILS

Metric	Value
Total Rows	4966
Columns	129
Attack	3866
Natural	1100

Dataset link https://www.kaggle.com/datasets/bachirbarika/power-system

4. MACHINE LEARNINGS CLASSIFIERS

K-Neighbors Classifier is a powerful tool for anomaly-based IDS in cyber-physical systems. This classifier works by comparing new data points to existing ones and classifying them based on the majority class among the nearest neighbours. It is effective for detecting anomalies because it identifies instances that significantly differ from normal behaviour patterns.

In the context of cyber-physical systems, K-Neighbour’s Classifier can quickly flag unusual activities that might indicate intrusions or faults. Its simplicity and effectiveness make it a valuable component of an IDS, especially when quick, reliable detection is essential for system security and stability. However, it may struggle with high-dimensional data and requires careful parameter tuning for optimal performance.

Random Forest Classifier is highly effective for anomaly-based IDS in cyber-physical systems. It operates by constructing multiple decision trees during training and outputting the mode of the classes for classification tasks. This ensemble method enhances accuracy and robustness, making it suitable for detecting anomalies within complex cyber-physical systems.

Random Forest Classifier excels at handling large datasets with high dimensionality and can capture intricate patterns indicative of anomalies. It is resistant to over fitting due to its averaging process and provides insights into feature importance, aiding in understanding the factors contributing to anomalies. Its robustness, accuracy, and interpretability make Random Forest Classifier a vital component for maintaining the security and integrity of cyber-physical systems.

XGBClassifier (Extreme Gradient Boosting) is a highly efficient tool for anomaly-based IDS in cyber-physical systems. It works by iteratively improving weak learners, specifically decision trees, through gradient boosting. This process optimizes performance by minimizing error and enhancing model accuracy. XGBClassifier excels in handling large, complex datasets and effectively captures subtle anomalies within cyber-physical systems. It provides high accuracy and robustness, making it ideal for detecting rare and sophisticated intrusion patterns. Additionally, it offers features like regularization to prevent overfitting and handles missing values well, ensuring reliable and comprehensive anomaly detection for maintaining system security and integrity.

Decision Tree is a straightforward yet powerful tool for anomaly-based IDS in cyber-physical systems. It operates by creating a tree-like model of decisions, where each node represents a feature, each branch a decision rule, and each leaf a class label. This method is intuitive and easy to interpret, making it useful for identifying and explaining anomalies.

In cyber-physical systems, Decision Tree classifier can quickly detect deviations from normal behaviour by analysing system data. While it is prone to over fitting, it performs well when combined with other techniques such as pruning or ensemble methods. Its simplicity and clarity make Decision Tree classifier a valuable asset for real-time anomaly detection and system security maintenance.

SGD classifier is efficient for handling large datasets due to its linear nature, but it may be less accurate compared to other classifiers. SVM (Support Vector Machine) excels in high-dimensional data, making it ideal for complex anomaly detection tasks. LGBM classifier (Light Gradient Boosting Machine) offers high performance and speed, handling large-scale data efficiently. AdaBoost classifier enhances weak models to achieve better results, improving detection accuracy. Bagging classifier increases stability by combining multiple models, reducing variance. MLP classifier (Multi-Layer Perceptron) leverages deep learning for capturing complex patterns, though it requires significant computational resources. Together, these classifiers ensure comprehensive and effective anomaly detection in cyber-physical systems.

Here are the formulas for precision, recall, f1-score, and support, which are essential metrics for evaluating the performance of classification models:

1. Precision:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{7}$$

Precision measures the accuracy of positive predictions, where TPTPTP is true positives and FPFPFP is false positives.

2. Recall:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{8}$$

Recall (or Sensitivity) measures the ability of the model to capture all relevant instances, where TPTPTP is true positives and FNFNFN is false negatives.

3. F1-Score:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

The F1-Score is the harmonic mean of precision and recall, providing a balance between the two metrics.

4. Support:

$$\text{Support} = \text{Number of actual occurrences of each class in the dataset} \tag{10}$$

Support is the count of true instances for each class in the dataset. It is not a measure of accuracy but provides insight into the distribution of the dataset.

These metrics are essential for understanding the performance of classification models, especially in imbalanced datasets where precision and recall offer more insight than accuracy alone.

TABLE III. COMPARISON OF ML CLASSIFIERS

Classifier	**Accuracy**	**Precision**	**Recall**	**F1-Score**
KNeighbors	0.8451	0.88	0.93	0.9
Random Forest	0.9588	0.96	0.99	0.97
XGB	0.9507	0.97	0.97	0.97
Decision Tree	0.9245	0.95	0.95	0.95
SGD	0.5765	0.75	0.69	0.71
SVM (Support Vector Machine)	0.7736	0.78	0.99	0.87
Light GBM (Light Gradient Boosting Machine)	0.9547	0.96	0.98	0.97
AdaBoost	0.8501	0.87	0.94	0.91
Bagging (SVC)	0.7817	0.79	0.98	0.87
MLP (Multi-Layer Perceptron)	0.7706	0.78	0.98	0.87

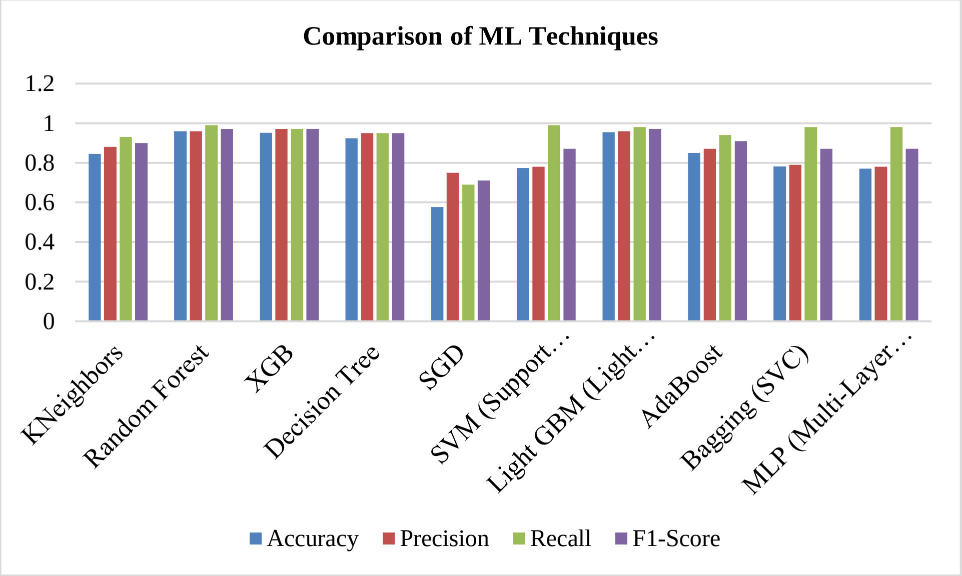

Fig. 4. Comparison of ML Techniques

Here is a summary of the classifier performance based on the provided metrics:

Random Forest achieves the highest accuracy (95.88%) and excels in all metrics with a precision of 0.96, recall of 0.99, and an F1-Score of 0.97, making it the most reliable and robust classifier for this dataset.

XGB and LightGBM also perform exceptionally well, with accuracies of 95.07% and 95.47%, respectively. Both classifiers show high precision (0.97 and 0.96), recall (0.97 and 0.98), and F1-Scores (0.97 for both), indicating their strong performance in detecting anomalies.

Decision Tree provides a balanced performance with an accuracy of 92.45%, and a high F1-Score of 0.95, demonstrating good precision and recall.

KNeighbors performs well with an accuracy of 84.51%, a precision of 0.88, and a recall of 0.93, though it slightly trails behind the top performers in F1-Score (0.90).

AdaBoost offers a solid performance with an accuracy of 85.01%, precision of 0.87, recall of 0.94, and an F1-Score of 0.91, indicating good overall effectiveness.

SVM and Bagging (SVC) show lower accuracy compared to the top classifiers but are competitive in recall (0.99) and F1-Score (0.87), with accuracy values of 77.36% and 78.17%, respectively.

MLP also has lower accuracy (77.06%) but performs well in recall (0.98) and F1-Score (0.87), similar to SVM and Bagging.

SGD has the lowest accuracy (57.65%) and performs less well in precision (0.75), recall (0.69), and F1-Score (0.71), indicating it may not be as effective for this task.

In summary, Random Forest, XGB, and LightGBM are the top-performing classifiers for this anomaly detection task, while SGD performs the least effectively.

5. CONCLUSION

In conclusion, for anomaly-based IDS in cyber-physical systems, Random Forest, XGB, and LightGBM excel in accuracy, precision, recall, and F1-Score, making them highly effective for detecting anomalies. These classifiers offer robust performance, handling complex intrusion patterns efficiently. Decision Tree and AdaBoost also provide strong results, though with slightly lower overall metrics. SVM, Bagging (SVC), and MLP achieve good recall and F1-Score but fall short in accuracy. SGD performs the least effectively, with lower accuracy and performance metrics. Thus, Random Forest, XGB, and LightGBM are recommended for their comprehensive anomaly detection capabilities in cyber-physical systems.

Conflicts Of Interest

The author's disclosure statement confirms the absence of any conflicts of interest.

Funding

The author's paper does not provide any information on grants, sponsorships, or funding applications related to the research.

Acknowledgment

The authors expresses appreciation to the institution for their continuous support and access to relevant research materials.

References

[1] G. Costa Silva, R. M. Palhares, and W. M. Caminhas, “A transitional view of immune inspired techniques for anomaly detection,” in International Conference on Intelligent Data Engineering and Automated Learning, 2012, pp. 568–577.

[2] S. Yuan and X. Wu, “Deep learning for insider threat detection: Review, challenges and opportunities,” Computers & Security, vol. 104, p. 102221, 2021.

[3] T. K. Das, S. Adepu, and J. Zhou, “Anomaly detection in industrial control systems using logical analysis of data,” Computers & Security, vol. 96, p. 101935, 2020.

[4] M. Bertolini, D. Mezzogori, M. Neroni, and F. Zammori, “Machine learning for industrial applications: A comprehensive literature review,” Expert Systems with Applications, vol. 175, p. 114820, 2021.

[5] W. L. Duo, M. C. Zhou, and A. Abusorrah, “A survey of cyber attacks on cyber physical systems: Recent advances and challenges,” IEEE/CAA J. Autom. Sinica, vol. 9, no. 5, pp. 784–800, May 2022, doi: 10.1109/JAS.2022.105548.

[6] T. Morris, “Industrial Control System (ICS) Cyber Attack Datasets,” [Online]. Available: https://sites.google.com/a/uah.edu/tommymorris-uah/ics-data-sets. [Accessed: Sep. 1, 2022].

[7] D. Shalyga, P. Filonov, and A. Lavrentyev, “Anomaly detection for water treatment system based on neural network with automatic architecture optimization,” arXiv preprint, arXiv:1807.07282, 2018.

[8] H. M. Saleh, A. K. Oleiwi, and A. A. H. Abed , Trans., “Detecting attacks in banks by cyber security: an applied study”, Babylonian Journal of Machine Learning, vol. 2023, pp. 65–72, Nov. 2023, doi: 10.58496/BJML/2023/011.

[9] J. Inoue, Y. Yamagata, Y. Chen, C. M. Poskitt, and J. Sun, “Anomaly detection for a water treatment system using unsupervised machine learning,” in 2017 IEEE International Conference on Data Mining Workshops (ICDMW), IEEE, 2017, pp. 1058–1065.

[10] M. Kravchik and A. Shabtai, “Detecting cyber-attacks in industrial control systems using convolutional neural networks,” in Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy, 2018, pp. 72–83.

[11] Z. Yu, Z. Kaplan, Q. Yan, and N. Zhang, “Security and privacy in the emerging cyber physical world: A survey,” IEEE Commun. Surv. Tutor., vol. 23, no. 3, pp. 1879–1919, 2021.

[12] Q. Zhang, A. Z. Mohammed, Z. Wan, J.-H. Cho, and T. J. Moore, “Diversity-by-design for dependable and secure cyber-physical systems: A survey,” IEEE Trans. Netw. Serv. Manag., vol. 19, no. 1, pp. 706–728, 2021.

[13] B. Barika, “Power system,” Kaggle. Available: https://www.kaggle.com/datasets/bachirbarika/power-system. [Accessed: Sep. 10, 2024].

[14] A. Chevrot, A. Vernotte, and B. Legeard, “CAE: Contextual auto-encoder for multivariate time-series anomaly detection in air transportation,” Computers & Security, vol. 116, p. 102652, 2022.

[15] A. Barbado, Ó. Corcho, and R. Benjamins, "Rule extraction in unsupervised anomaly detection for model explainability: Application to one-class SVM," Expert Systems with Applications, vol. 189, p. 116100, 2022.

[16] R. C. Borges Hink et al., "Machine learning for power system disturbance and cyber-attack discrimination," in 2014 7th International Symposium on Resilient Control Systems (ISRCS), Aug. 2014, pp. 1–8, doi: 10.1109/ISRCS.2014.6900095.

[17] G. Ravikumar and M. Govindarasu, "Anomaly detection and mitigation for wide-area damping control using machine learning," IEEE Trans. Smart Grid, 2020.

[18] G. Zizzo, C. Hankin, S. Maffeis, and K. Jones, "Intrusion detection for industrial control systems: Evaluation analysis and adversarial attacks," arXiv preprint, arXiv:1911.04278, 2019.

[19] D. Li et al., "MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks," in ICANN, 2019.

[20] W. Schneble and G. Thamilarasu, "Attack detection using federated learning in medical cyber-physical systems," in 28th International Conference on Computer Communications and Networks (ICCCN), 2019, pp. 1–8.

[21] S. Rajaprakash, C. B. Basha, M. Nithya, K. Karthik, N. Aggarwal, and S. Kayathri , Trans., "RNN-Based Framework for IoT Healthcare Security for Improving Anomaly Detection and System Integrity ", BJIoT, vol. 2024, pp. 106–114, Oct. 2024, doi: 10.58496/BJIoT/2024/013.

[22] Z. Zhang et al., "Privacy-enhanced momentum federated learning via differential privacy and chaotic system in industrial cyber-physical systems," ISA Trans., Sep. 2021, doi: 10.1016/j.isatra.2021.09.007.

[23] S. Tuli, G. Casale, and N. R. Jennings, "TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data," arXiv preprint, arXiv:2201.07284, May 2022, doi: 10.48550/arXiv.2201.07284.

[24] A. Mutar Awad et al., "A comprehensive review on anomaly detection of man-in-the-middle attacks in cyber physical power systems," in 3rd Annual Int. Conf. on Information and Sciences (AICIS'23), Fallujah University, 2023.

[25] A. Mutar Awad et al., "Model-based anomaly detection in cyber physical power systems," AIP Conf. Proc., vol. 3207, no. 1, p. 070001, Sep. 2024, doi: 10.1063/5.0234148.

[26] W. Schneble and G. Thamilarasu, "Attack detection using federated learning in medical cyber-physical systems," in 28th International Conference on Computer Communications and Networks (ICCCN), 2019, pp. 1–8.

[27] A. S. . Bin Shibghatullah, "Mitigating Developed Persistent Threats (APTs) through Machine Learning-Based Intrusion Detection Systems: A Comprehensive Analysis", SHIFRA, vol. 2023, pp. 17–25, Mar. 2023, doi: 10.70470/SHIFRA/2023/003.

[28] Z. Zhang, L. Zhang, Q. Li, K. Wang, N. He, and T. Gao, "Privacy-enhanced momentum federated learning via differential privacy and chaotic system in industrial cyber–physical systems," ISA Trans., Sep. 2021, doi: 10.1016/j.isatra.2021.09.007.

[29] S. Tuli, G. Casale, and N. R. Jennings, "TranAD: Deep transformer networks for anomaly detection in multivariate time series data," arXiv preprint, arXiv:2201.07284, May 14, 2022, doi: 10.48550/arXiv.2201.07284.

[30] A. M. Awaad, K. M. A. Alheeti, A. Kream, and A. H. Najam, "A comprehensive review on anomaly detection of man-in-the-middle attacks in cyber physical power systems," Accepted for publication in 3rd Annual Int. Conf. on Information and Sciences (AICIS'23), University of Fallujah, 2023.

[31] A. M. Awad, K. M. A. Alheeti, and A. K. A. H. Najem, "Model-based anomaly detection in cyber physical power systems," AIP Conf. Proc., vol. 3207, no. 1, p. 070001, Sep. 2024, doi: 10.1063/5.0234148.