[1]S. R. Kotecha
[2]R. J. Khimani
[3]R. J. Trivedi
[4]P. D. Maheta
[5]H. M. Rathod
[6]C. R. Varnagar

Evaluation of Classifiers to Detect Intrusion in SCADA System

Journal of Electrical Systems

Abstract: - Critical infrastructures play a important role in bringing the economy on track, where infrastructures such as – smart grids, gas pipelines, nuclear power stations, and water pumps controlled and managed by Supervisory Control and Data Acquisition (SCADA) systems, which is a crucial entity in Industrial Control System (ICS). One of the major initiatives that had been taken is to connect these ICS systems with the Internet, although it brings a lot of challenges too related to cyber security such as –unmonitored communication, un-encrypted network traffic, weak protocols, external attack – intruders, lack of asset awareness. The information exchange between Programmable Logic Controller (PLC) and SCADA system can be prone to different cyber attacks, therefore a detection mechanism is very essential. An automated system such as Intrusion Detection System (IDS) or Intrusion Prevention System (IPS) would be very supportive of any machine learning algorithm, in particular to the classification problem. This paper would mainly focus on diverse classification algorithms to increase the efficiency of intrusion detection. The techniques of newly adopted classification algorithms such as – Linear Discriminate Algorithm (LDA), and Quadratic Discriminate Algorithm (QDA), have been investigated thoroughly, and applied to the SCADA gas pipeline dataset. The performances of individual algorithms are compared and evaluated closely in terms of accuracy, precision, and recall. True-positive rates of the algorithms are taken into consideration while comparing their performances.

Keywords: ICS, Machine Learning, SCADA, PLC, Cyber security

I. INTRODUCTION

Industrial Control System (ICS) are the control system that are associated with instrumentation which includes – devices, networks and controls used to operate or automate several industrial processes. There are two well-known ICS systems namely – Supervisory Control and Data Acquisition (SCADA) and Distributed Control System (DCS) which are widely adopted in the industry to monitor and control different processes of water pump stations, oil and gas pipelines, electric power grids and nuclear stations. A typical SCADA system consists of Human Machine Interface (HMI), Supervisory System, Remote Terminal Unit (RTU), Programmable Logic Controller (PLC) and Communication Infrastructure shown in Figure 1. It has a centralized system whose primary purpose is to long distance control and monitoring of field sites which are located at different places. The SCADA network connects to different hardware and software cumulatively called as – Operational technology (OT) which will observe physical devices in the field. Several PLCs are connected to the network whose work is start and stop any process for example – start and stop motor conveyor belts, record and monitor temperature, pressure and trigger alarms during any emergency situation. HMI is a graphical user interface which allows a human to interact with any controller hardware; it will display data and status of any control devices, monitor and configure various set points, control algorithms and parameters. There are MTUs connected to the network which will issue commands to PLCs or RTUs (microprocessor control field device). The architecture has Intelligent Electronic device (IEDs) which are smart devices competent in gaining data, local processing and communicating with other IEDs and RTUs. A centralized database is managed by historian who has all logging information related to any process, used in process analysis, statistical process control and organizational level planning.

[1, 2,3,4,5,6] Assistant Professor, Computer Engineering Department, Government Engineering College, Rajkot, Gujarat, India.

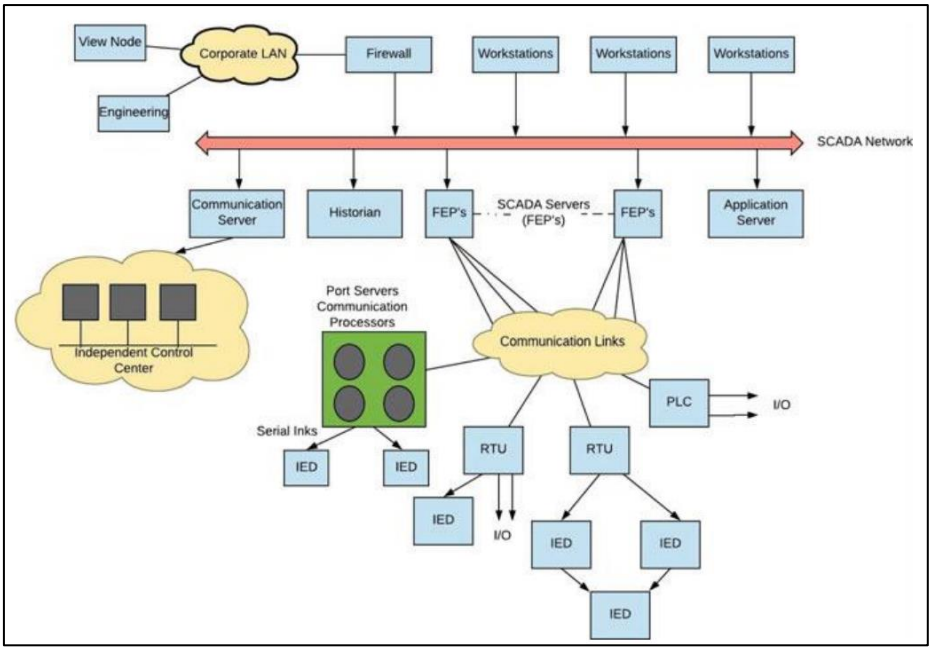

Figure 1. SCADA Architecture consists of different PLC, historian, communication system with master and slave devices using varied SCADA protocols.

The communication between different devices, servers and workstation are done by proprietary protocols. Process Field Bus (PROFIBUS) used in between RTU to MTU, MTU to MTU and RTU to RTU communication. It is further classified in Profibus Decentralize Peripherals (Profibus DP) operate in sensors and actuators. Profibus Process Automation (Profibus PA) used to monitor and measure equipment's through a control system. Distributed Network Protocol (DNP3) operates at layer 2, layer 4 and layer 7 of OSI model, generally used in electricity, and wastewater treatment plants. Modbus is one of the oldest protocols, uses serial communication with PLCs. It has two version – Serial Modbus purposes for standard data transmission with High Data Link Control (HDLC) and Modbus TCP uses TCP/IP protocol stack to transmit data. [1] has simulated SCADA network using virtualization, where they have infected master with malware attack which will flip the bits of function code for reading coil. Another man in the middle attack has been performing between master and slave, where ARP tables get poisoned; this will force the network to trust this new route for communication. The traffic between master and slave now captured and modified by Scapy tool. Open Platform Communication (OPC) standard is based on Microsoft windows operating system which ensures seamless flowing of information among devices. Building Automation and Control Networks (BACnet) is one more communication protocol which will regulate building heating, ventilating and air conditioning, building access and fire detection. Due to SCADAs large-scale deployment, it is integrated in IP network where it is exposed to various cyber attacks. These networks are accessible via remote connections and there is a wide possibility that an attacker can access one of the line or an intranet which is connecting local network to the outside world. It has been observed that most attack happen in SCADA system are of intrusion attacks which leads to other possible attacks on Historian, PLC, servers and access control systems, shown in Table **1.** Moreover, SCADA uses weak protocols and does not support – cryptography, that leads to another threat in communication – sniffing. Due to the increase rate of cyberattacks on control systems, specific actions have been taken to increase the security of SCADA networks. Department of Energy, US has provided 21 steps to improve the security of SCADA network [2] – Implement internal and external IDS, technical audits, Network protection strategy, disaster recovery and system backup plans are some of the focused points.

In this paper, we discussed several shortcomings of SCADA system with its countermeasures, for-instance – Weaken protocols, poor access control, inexact firewalls and ineffective encryption standards. Each shortcoming has its own way to encounter an attack. This paper provides machine learning (ML) to defend SCADA system. ML and Deep Learning (DL) algorithms had competent power to classify the network traffic and hence we can embed such classifiers into our detection systems. In fact, tuning the weights of these classifiers can increase the accuracy of categorization. Section 4 discuss ML algorithms with its strengths and weakness. Furthermore, we

have used WUSTL-IIOT-2018 dataset which has 1048575 x 7 datapoints, and gives output as a binary classifier – "0" normal traffic and "1" attack traffic. We have applied classifier such as – ensemble methods, tree, discriminant analysis and logistic classifiers on this dataset. Prior to that, we have balanced the dataset using resampling techniques. Ensemble technique has given a better accuracy compared to others, evaluated using metrics – Precision, Accuracy, f1-score, MCC etc.

The rest of the paper is organized as follows, Section 1 – introduces SCADA Architecture, Section 2 – highlight distinct security challenges of SCADA system with its aid and related work in this domain. Section 3 – represents related work. Section 4 - presents data analysis of the dataset and machine learning algorithms. Section 5 – deals with results and analysis. We conclude our work and future scope in Section 6, 7.

II. SCADA SECURITY CHALLENGES

There is now an emergent need in strengthening industrial control system from cyberspace, which has put some major challenges in securing such systems. This section will provide all major security challenges faced by SCADA network and we will discuss some of its technical solutions which has directed by the security community. A study conducted by Raytheon and Ponemon Institute, where they interviewed 1,000 IT professionals from U.S, Europe and the Middle East/North Africa, to rate which cyber attack would be increase in frequency in due time. It has been observed, on an average attack on SCADA network will increase by 15% from current 40% which means several control and supervisory system are going to be manipulated by attackers which can risk many lives. Below few subsections will discuss several loopholes of SCADA network and how we can improve their security.

Attack Methods	Target Vectors								
	Access Control	Application Server	PLC	Data Historian	Master/Slave devices	HMI	Firewall /IDS/IPS	SCADA Server	Communication System
Unpatched Application	✓	✓		✓	✓	✓		✓	
Unprotected Network Access Points	✓						✓	✓	✓
Remote Access	✓	✓	✓	✓	✓			✓	
Insecure Implementation of Protocols	✓	✓		✓	✓		✓	✓	
Malware Installation		✓	✓	✓	✓	✓		✓	
Database Injection		✓		✓				✓	
External Intrusion	✓		✓				✓		
Reply Attack			✓						
DoS Attack			✓					✓	✓
DNS Spoofing							✓		✓
Firewall Exploitation				✓			✓		✓
Exploitation of Login Credentials	✓	✓		✓			✓	✓	
Phishing									✓
Exploitation of Vulnerable	✓	✓	✓	✓	✓			✓	

Protocol									
Buffer Overflow					✓				
Expose Public Private Key	✓		✓	✓				✓	✓

Table 1. Attacker's attack methods which can be used on SCADA components as target vectors. The common target vectors are SCADA, Application Server and Access Control.

2.1 SCADA Weaken Protocols

Protocols are there to make flawless communication between several components of SCADA such as PLCs, RTUs and MTUs, however these protocols are having several vulnerabilities which give a scope to attacker to attack. Table 2 will list down all SCADA protocols with their shortcomings. In [3] authors bring security awareness in one of the most recent protocol used in SCADA – PCOM. They dissect it, and find out several vulnerabilities in the protocol which can leverage by an attacker to gain full control on the controller. They analyzed the protocol by dissecting it using Wireshark PCOM dissector, created Nmap scripts and Metasploit modules to further understand PCOM messages. Several attacks have been performed on PCOM protocol and test it with Snort rules for further assessment.

2.2 Access Control

SCADA networks are not secure with unauthorized access, any internal and external intrusion can happen in SCADA network. It is decisive to improve the access control mechanism, unfortunately, it is a major challenge to create a perimeter for access control, as such networks are not only connected to outside world by gateways, but has other means of communication such as telephones lines. These connections can be easily lured to make a prohibited access inside a secure network. Gateways are the means by which a network can communicate with other networks, these devices are not having any security features, it is advisable to create security enabled gateways. Furthermore, refine access control solutions, define new network security policies and two-way or smart card authentication is needed, as simple password-based authentication induces to social engineering attack – phishing.

2.3 Firewall and Detection System

Protocol	Security Concern	Possible Countermeasures
Modbus [6]	Lack of Authentication – Modbus session requires only modbus address, function code and associate data, results in MitM and replay attacks. Lack of Encryption – modbus commands and addresses are transmitted in cleartext. Broadcasting – serially connected devices will receive all data; DoS attack can happen.	Proper Function Scan, Deep Packet Inspection (DPI) for protocol filtering [8]. Whitelist of devicesneed to be created. Validate modbus session using any application data monitor.
DNP3 [7]	Lack of Authentication and Encryption advance to spoof the master node which can control the slaves – capture token, inject false taken.	Proper Hardening of System. Rigorous System Assessments and Auditing. Patched any unpatched interconnections.
PROFIBUS [9]	Lacking physical security, authentication – Stuxnet compromised PLC (works on PROFIBUS)	Proper Access control.
EtherNET/IP	Messages uses UDP protocol – no reliability, sequencing and integrity checks.	Application layer inspection. Better firewall and ICS-aware IPS can be used.

PROFINET	Allowing any vulnerabilities of Ethernet and IP.	Use in a secure authenticate and encrypted network only. Firewall and ICS-aware IPS can be used.
EtherCAT [10]	Highly sensitive to DoS attack, Over UDP no mechanism to check integrity, reliability or sequencing.	Passive network monitoring, ICS aware IPS to restrict traffic from unauthorized source. MAC address filtering is necessary to protect EtherCAT devices from an external attack.

Table 2. SCADA protocols with its security concern and countermeasures

Firewall is a utility, hardware or software to monitor network traffic. It has the capability to deny or accept the traffic based on the rule it is configured to. In addition to that, it configured to control and monitor any unauthorized activities, deny and allow traffic based on specific protocol. For example – we can set firewall rules again PROFINET protocol to make its traffic allow or deny in local or remote SCADA network. [4] authors have designed a new SCADA firewall – SCADAWall, aims to prevent SCADA network from threats with three algorithms – Comprehensive Packet Inspection(CPI), Proprietary Industrial Protocols Extension Algorithm (PIPEA) and Out-of-Sequence Detection Algorithm (OSDA). The performance of all algorithms is compared and evaluated; CPI shows compulsive results in real time environment without sacrificing network performance. The firewall has its limitation such as it is not that efficient in dynamic situation, creating rule for such situation is a major challenge. SCADA firewalls can also be improved by developing micro-firewalls that can be embedded in each SCADA device. Hence firewall can rule out other device traffic and accept only the traffic which is intended for himself.IDS/IPS on the other hand, is relatively incapable to detect any suspicious behavior of SCADA protocols, because the rules an IDS/IPS needed for detection, need a rigorous vulnerability assessment of protocols.

2.4 Encryption and Key Management

SCADA key management has two categories – centralized and decentralized architectures. Centralized key distribution – Key distribution center (KDC) shared secret key for secure communication between different nodes, whereas in decentralized architecture, there is no trusted KDS which can share secret key, instead it is established by pre-shared master key, which allows session between nodes. [5] shows a review of different key management formed by distinct researchers. They have used well-defined parameters to compare and evaluated these key management systems based on – Baud rate and Storage cost of keys. Hybrid asymmetric and symmetric key management is needed to secure the session keys, generic keys between nodes.

III. Related Work

This section outlines numerous survey and literature review of SCADA system, it talks about various classification and prediction methods of machine learning. SCADA system were never designed for network communication or security, its purpose was to provide reliable services and provide security using proprietary standards [11],[12]. But as the demands of automation and Third-party vendor – Commercial-off-the-Shelf (COTS) – any application or service built by third party vendor, get increases, control systems get integrated with network communication. As a result, attackers got a widespread area to explore and attack. Since 1990's there are seamless attacks occurred on control system [13] shows us security schemes on current standards, attack detection and prevention mechanism, [14]. It is not only one interface of network from where attacker can access the control system, several components are connected to wired, wireless, or cellular networks. Most of the SCADA protocols through which every component can communicate with each are weaken and just talks in plain-text. It is easier to perform any MiTM attacks. [15] authors used emulated testbed of a real water – EU Project FACIES [16], and carry out Modbus flooding against one PLC. It sends 100,000 fake read input registers queries from SCADA to PLCA, which lasts 10s. Another MITM attack they performed, where they modified one Modbus/TCP packet. The operator was unable to close valve 1 to deactivate Pump 2, as its content

were intercepted and modified. The situation may get worst, leads to bad consequences where valve cannot be closed and water level in tank is increasing – overflow can occur. Security practitioner uses simulation [17] to study more about such automated system, due system dependencies it is hard to create such simulations or testbed. Hence, it is growing field to study to create strict standards and communication protocols in order to defend from an attack [18].Authors of [19] has incorporated machine learning in SCADA system, where they have used gas pipeline SCADA dataset [20],[21] which has 274628 instances. They study imbalance nature of dataset and resolved using cost-sensitive learning and fisher's discriminant analysis. The increase the accuracy of the algorithm they have applied several ML algorithms wherein RandomTree substantially outperforms. Accuracy, False Positive Rate, True Positive Rate, F1-Score and MCC are the evaluation metric used for the analysis. Random Forest gives 97.8% accuracy and 93.5 MCC.[22] detected anomaly in EtherCAT with machine learning algorithms, for that they have created water level control system testbed. They have generated 16 different events grouped in four categories, which includes normal and abnormal behavior. Several machine learning algorithms have been adopted for the prediction, among them KNN has given better accuracy (100%). We can observe, that many literature shave their own testbeds which allow them to create datasets, our paper has this limitation, we have to depend on other's dataset. Since, it is their dataset, we cannot customize it the way we want to make our model learn. When working with such critical infrastructure, the importance of testbed cannot be ignored [23]. According to [24],[25], specifically for SCADA security testbeds we can use any of the following way to implement it.

- Combine real hardware with software to do actual experimentation, though it is very expensive, but it gives real-world experience and the solutions we got can be adopted in the real infrastructure – Cyber Physical Testbeds [26].
- Use of physical devices in a software (emulation), with that software you can process and control the device [27].
- Simulation based testbed [28] – We can use software such as Omnet++, MATLAB etc to do the simulation of system (gas pipeline, water tank system). Moreover, this also empathizes to use virtualization, where several machines can become individual component of SCADA system [29].

Communication used by ICs is different from regular IT systems, it is more prevalent to the cyber-attacks. There are researchers who are working in ML based detection system [30], which can solve the vulnerabilities related to Integrity, Confidentiality and Availability. Authors of [31] has created real-world test of water tank system, which has several other components – HMI, IDS, PLC and Firewall, communicating with each other using Modbus TCP and ethernet. They deliberately built their dataset as imbalanced, assuming the testbed works similar to possible real-world ICS. They applied several algorithms to validate their models (Random Forest, Logistic, SVM, naïve bayes etc.). Random Forest has given better accuracy among them.

IV. Machine Learning Algorithms to Defend SCADA Networks

In this section, we will incorporate machine learning (ML) algorithms to find cyberattacks, which can be further help in improving our firewalls, IDS, and IPS. It is relatively difficult to create control system testbed because of its complex architecture, fairly we can simulate these systems and collect data for ML learning. Hanyu [32] show the review of many learning algorithms to its fault detection in the power system. The algorithms include conventional, deep, reinforcement and training learning. [33] convey a vast literature of SCADA based intrusion detection system. Literature shows research work on SCADA IDS in the period 2015-2019. They proposed an IDS which works particularly on machine learning using open-source tools and creating sophisticated ICS testbeds. In addition to that, this section also surveys several machine learning classifiers, algorithms and its strength and weakness.

4.1 Ensemble Methods

The ensemble model takes decision from other models to improve the overall performance [34]. The idea behind this approach is to take responses from generalized and diversified models and combine it to have a cumulative response. It turns out that, it is preferable approach to get decision as compared to individual models. Examples of ensemble classifier – AdaBoost, Bagging, ExtraTree, GradientBoosting, RandomForest.

- Adaboost Classifier

Adaptive boosting (adaboost classifier) is an ensemble technique [35],[36], which makes n number of decision tree during its training. Each tree model noted down its errors and the incorrect classification passed to the next

model; this process gets continued till specific conditions met. The intention is to lower the weighed error (E_t), it can be done by assigning higher weight ($w_{i,t+1}$) to the observation which have been predicted incorrectly.

Algorithm: AdaBoost Classifier:

A sample data consists of $(x_1, y_1) \ldots (x_m, y_m)$ where $x_i \epsilon$ X as inputs, $and\ y_i\ \epsilon \{-1, +1\}$ as desired output.

Initialize weights $w_i = \frac{1}{m}$ {where, $i = 1 \ldots m$}

Weak Learners ($h : x \rightarrow \{-1, +1\}$)

For $t\ =\ 1..T$:

- Train weak learners $h_t(x)$ using weight w, which minimizes weighted sum error.

$$E_t = \sum_{\substack{i=1 \\ h_t(x_i) \neq y_i}}^{m} w_{it}$$

where, $h_t(x_i)$ is a weak learner.

- Get Hypothesis $H(x) \epsilon \{-1, +1\}$
- Aim: Select weak classifier (h_f) with lower weighted error.
- Choose $\alpha_t\ =\ \frac{1}{2}\ ln(\frac{1-E_t}{E_t})$
- Adding it to Ensemble and update the weights for $i\ =\ 1 \ldots . m$.

$$w_{i,t+1} = w_{i,t} exp^{-\alpha_t y_i h_t(x_i)}$$

Normalize it using constant Z_m, such that,

$$w_{i,t+1} = 1$$

Final Hypothesis:

$$H(x)\ =\ sign\ (\sum_{t=1}^{T} \alpha_t h_t(x))$$

- Bagging

Bootstrap aggregating is another ensemble algorithm [37], which reduces the overfitting by decreasing the variance from the dataset. The idea is to create bootstrap dataset (B) by randomly selecting samples from original training sample, with replacement. Apply any classifier (Decision tree mostly) to predict new samples (P_i). With major voting (average) of new predicted class, the most often predictor will be the output [38],[39].

Algorithm: Bagging

Given:A sample Dataset have training set (D).

For $1..m$:

- Create D' (bootstrap Dataset) from original dataset (D).
- Classifier (C_i)= will be built from each set of D' with replacement.

C_i will predict the class $(P_1 \ldots P_m)$

Return the class that often predicted using voting,

$$C_i(x)\ =\ argmax\ \Sigma(P_1 \ldots P_m$$

- Random Forest

Random forest is consisting of large number of deep trees which act as an ensemble method. Each tree will split out class prediction, with voting most often predictor gets our model prediction [40], [41].

Algorithm: Random Forest

Given: A training set (X) = $\{x_1 \ldots . x_n\}$, with output (Y) = $\{y_1 \ldots . y_n\}$.

Apply ensemble method – Bagging:

For $b\ =\ 1\ \ldots B$:

- Select n random sample from $(X, Y) -> \{X_b \ldots . Y_b\}$.

- Fit tree (using classification/regression) on sample $\{X_b \ldots . Y_b\}$
- Classifier (C_i) will predict the class by voting,

$$C_i = \frac{1}{B}\sum_{b=1}^{B} C_i(x^i)$$

Where, x^i = unseen sample, B = Bootstrap sample

- The process will decrease the variance without increasing bias, hence reduces $overfitting
- Random subset of features will be selected from the learning process.From Above selected features, if two or more features have strong predictors towards response (Y). it can also be found in other B trees, as a result we can find some correlation between such B trees. This signifies we can grow such tree to larger deep trees.

4.2 Logistic Classifier

Whenever we have binary classification problem, logistic regression [42] is the first classifier which comes in our mind, that outputs Y ∈ {1, 0 }. Linear regression is not a suitable option to this problem (though it depends on the dataset). At some point it fits well and sometimes not. Logistic regression needs an output in between 0 and 1. $0 \leq H(x) \leq 1$, where $H(x)$ is our hypothesis. Instead of using a line to fit all the data, we can just bend this line a little from both ends. This intuition leads to logit functions or sigmoid function which outputs in between 0 and 1. In this case the H(x) can be defined as,

$$H(x) = g(Z)$$

$$g(Z) = \frac{1}{1 + e^{-z}}$$

$$H(x) = g(w^T x)$$

Where, w^T is a transpose vector of all coefficient of input vector (x).

$$H(x) = \frac{1}{1+e^{-w^T x}} \quad \text{as.} g(Z) = g(w^T x)$$

This hypothesis can help us to achieve binary classification, we predict either (y=1) or (y=0).

Predict "y=1" if $H(x) \geq 0.5$

Predict "y=0", if $H(x) < 0.5$

4.3 Stochastic Gradient Descent (SDG) Classifier

SDG works on the principle of Gradient Descent; it is utilized to minimize the cost function of regression or classification problem. Logistic regression cost function $J(w)$ is defined as,

$$J(w) = \frac{1}{m}\sum_{i=1}^{m} Cost\ (h(x)^i, y^i)$$

$$Cost(h(x), y) = -log\big(h(x)\big), \quad if\ y = 1$$

$$-\ log\big(1 - h(x)\big),\ if\ y = 0$$

$$J(w) = \frac{-1}{m}[\sum_{i=1}^{m} y^i\ log(h(x)^i) + (1 - (y)^i)\ log\ (1 - h(x)^i)] \quad (1)$$

To minimize the cost function, we can run a gradient descent,

Repeat until convergence:

$$w_j = w_j - \alpha \frac{\delta}{\delta_j} J(w)$$

Simultaneously update all w_j

Replace Eq(1) in Eq(2).

$$w_j = w_j - \alpha \frac{\delta}{\delta_j} \left(\frac{-1}{m} \sum_{i=1}^{m} (y^i \, log(h(x)^i) + (1-(y)^i) \, log \, (1-h(x)^i)\right)$$

We will iterate it, till w_j gets minimizes. $J \epsilon \{1,2 \ldots . n\}$, h is a sigmoid function and α is learning rate. The drawback of this approach is its iteration. For instance, if we run gradient descent on a larger dataset, it will take all datapoints (x_i) and calculate derivativesto minimizes the coefficients. This convergence needs huge number of iterations, which is computationally expensive. There comes the stochastic gradient descent which will randomly picks the datapoints from the whole dataset at each iteration, and reduces the computation.To reduce the overfitting of the model, we can regularization – Lasso or Ridge Regularization.The logistic regression with this regularization is defined as follows.

$$J(w) = \frac{-1}{m} \sum_{i=1}^{m} (y^i \log (h(x)^i) + (1-(y)^i) \log (1-h(x)^i)) + \frac{\lambda}{2m} \sum_{j=1}^{n} w^2$$

The above equation has Regularization term $\frac{\lambda}{2m} \sum_{j=1}^{n} w^2$, this will minimize the magnitude of all high degree coefficient (reducing variance).

4.4 Naïve Bayes Classifier

It is another classification algorithm, which is based on Bayes' Theorem. It assumes that the features are independent of each other, there is no correlation among them, that is the "naïve" in the classifier. This assumption makes him a proficient algorithm [43]. Bayes' Theorem states – Given a class (C_i) and Feature (X_i) – "The relationship between the probability of class before getting the feature and probability of class after getting the feature is [44]"

$$P(C_i \mid X_j) = \frac{P(X_j \mid C_i) * P(C_i)}{P(X_j)} \quad (3)$$

Bayes' Theorem Proof:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$
$$P(B|A) = \frac{P(B \cap A)}{P(A)}$$

Since, right hand side equations are equal to each other, as they are equal to P(A|B), we can write,

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)}$$

P(A|B) is a posterior probability, P(B|A) is a class conditional probability also known as likelihood. P(B) is a normalize constant. Hence, {P(A|B) is proportional to P(B|A) *P(A)}. Since, there is an assumption of feature independency, probability of getting the features values $P\,(x_j^1 = a_1, x_j^2 = a_2, \ldots . x_j^n = a_n | C_i)$ is equal to multiplying all individual probabilities.

$$P(x_j^1 = a_1|C_i) * P(x_j^2 = a_2|C_i) * \ldots . P(x_j^n = a_n|C_i) = \prod_k p\left(x_j^k = a_k|C_i\right)$$

Output of the naïve classifier is to select the class for which the below equation (4) gives maximum output.

$$P(C_i)\prod_k p\left(x_j^k = a_k|C_i\right)$$

A variant of naïve bayes is Bernoulli Naïve bayes [45]. If your dataset is more dominant with discrete values for instance – yes or no, true or false, success or failure, 1 or 0 etc. Bernoulli's would be a better choice; it uses Bernoulli distribution for feature values.

$$p(x) = P[X \rightarrow x] \Rightarrow q = 1 - p \;\; if\; x = 0$$
$$p, \qquad\qquad if\; x = 1$$

$P(x)$ is a probability distribution of feature value (x), For a random instance X, probability of success is given by $p(if\; x = 1)$, probability of failure is given by $q = (1-p), if(x = 0)$.

4.5 Linear Discriminant Analysis

Principal Component Analysis (PCA) [47] and Linear Discriminant Analysis (LDA) are the linear methods for dimensionality reduction. PCA provides better data representation in lower dimension, however this representation is not useful in data classification as data are not that separable [46]. LDA uses the same projection concept as PCA does, but it preserves the direction for data classification. General steps in LDA are as follows,

We have d dimensional sample $x_1 \ldots . x_n$, and 2 classes (C1 and C2)
Project all sample xi onto a line $r = r^t x_i$
Compute mean of projection $(\overline{u_1}\&\bar{u}_2)$ and mean of class 1 and $2(u_1, u_2)$.
Measure the separation between projection of different classes, $|\overline{u_1}\&\bar{u}_2|$
Compute the scatter matrix,
Scatter Matrix of original sample *Xi*before projection,

$$S_1 = \sum_{x_i \in C_1} (x_i - \mu_1)(x_i - \mu_1)^t$$
$$S_2 = \sum_{x_i \in C_2} (x_i - \mu_2)(x_i - \mu_2)^t$$

Scatter Matrix for projected samples,

$$\bar{s}_1 = \sum_{y_i \in C_1} (y_i - \overline{u_1})^2$$

$$\bar{s}_2 = \sum_{y_i \in C_2} (y_i - \overline{u_2})^2$$

Effective separation J(r), find direction of r which makes J(r) large, hence make better separation.

$J(r) = \frac{(\bar{u}_1 - \bar{u}_2)^2}{\bar{s}_1 + \bar{s}_2}$(5)

Compute class scatter matrix (S_c),

$$S_c = S_1 + S_2$$
$$y_i = r^t x_i$$
$$\bar{u}_1 = r^t(\mu_1)$$
$$\bar{s}_1 + \bar{s}_2 = (y_i - \overline{u_1})^2 + (y_i - \overline{u_2})^2$$
$$\bar{s}_1 + \bar{s}_2 = r^t S_c r \quad (6)$$

Compute between the class scatter matrix (S_b),

$$S_b = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^t$$

$$(\bar{u}_1 - \bar{u}_2)^2 = (r^t\mu_1 - r^t\mu_2)^2$$

$$(\bar{u}_1 - \bar{u}_2)^2 = r^t S_b r \qquad (7)$$

Place Eq (6) and (7) in Eq (5).

$$J(r) = \frac{r^t S_b r}{r^t S_c r} (8)$$

Calculate the derivate of $J(r)$= 0, with respect to,r, you will get eigen values.

$$S_b = \lambda S_c r$$

Optimal line direction of r, once you inverse this scatter matrix composed of eigen vectors and its eigen values, which indicates which direction has more information (important features).

$$\mathrm{r} = s_c^{-1}((\mu_1 - \mu_2)$$

The above algorithms and its variant are some of the famous Classifier model, which have been taken into consideration in our binary classification problem. Each individual model has their own strength and weakness, upon which we can tune it to get better result. Table 3 shows strengths and weaknesses of models which can be helpful to other researchers to review them and use it accordingly.

Classifier Algorithm	Strengths	Weakness
AdaBoost	Need few parameters to implement it. Provides Feature Selection	Sensitive to Outliers. Not appropriate to Imbalanced Dataset.
Bagging	Reduces Overfitting. Perform well in high dimensional data.	Computationally expensive (as each model have training algorithm and activation function). High bias, if properly not implemented.
Random Forest	Reduces Overfitting, improve accuracy. Used in classification and regression problem.	More complex, as tree gets large. Too much Variability in the method.
Decision Tree	No Pre-Processing needed. Handles Collinearity.	Overfitting, if keep building trees. Prone to outliers.
Gradient Boosting	No Pre-Processing needed. Provides several tuning options – hypertuning.	Computationally expensive, due to large number of trees.
Logistic Regression	Easy implementation and interpretation.	No. of observation < No. of features leads to overfitting.
Stochastic Gradient Descent	Computationally fast, as it picks random sample. Converge fast with large dataset.	Due to frequent changes, steps to reach minima gets noisy.
Naïve Bayes	Fast.	Zero frequency problem.

	If input has categorical data, it performs better (as fast decision can be made).	It assumes – features are independent, but in real life, they are not.
Discriminant Analysis	Simple implementation as Decision boundary is linear. Reduces dimensionality.	In high dimensionality, it uses many parameters. Linear decision boundary is not separating the class sufficiently.

Table 3: Strengths and weakness of several classifier modes.

It is difficult to find publicly-accessible ICS dataset for evaluating and increasing their efficiency and detection rate. [48] authors use 15 smart grids datasets, where they lodge an effective detection mechanism using Correlation based feature selection (CFS) and K-Nearest Neighbor (KNN) which improves the accuracy and reduces the computational overhead on smart grids. CFS will help them to reduce the number of features which will further used in classification, increasing detection accuracy [49]. From 128 network features they reduces it to 17 and achieves highest detection rate. For our study, we have taken an updated SCADA dataset, which has been created in 2018 by Teixeira and Salman [50]. They have created SCADA cybersecurity testbed, to emulate real-world control system and carry out pragmatic attack. These network- based attack can capture and monitored using Audit Record Generation and Utilization System (ARGUS) tool. The dataset has 1048575 instances and 7 features (authors have not mentioned on what basis they have taken these 7 features), where normal traffic represents by 0 and attack traffic represent by 1. There are 721827 normal traffic instances and 326748 attack traffic instances. The dataset has its drawback, it is an imbalanced dataset as it is more biased towards normal traffic, hence the detection rate will always be little high. To evaluate this imbalance data, we have used several metrics such as F1 Score, ROC-AUC, Precision, Recall and MCC.We have applied different classification algorithms to the dataset to check its accuracy in order to detect the attack.The dataset has 'Sport' - source port as one of the features, the highest number of ports which have been used with its kernel density function (KDE) is shown in Figure.2. Dynamic ports are used for attack, their ranges are in between 49152-65535. 65578 times port number 50963 used for attack traffic, though these ports doesn't show any services related to it, hence we cannot analyze the type of attack, which is another disadvantage of this dataset.

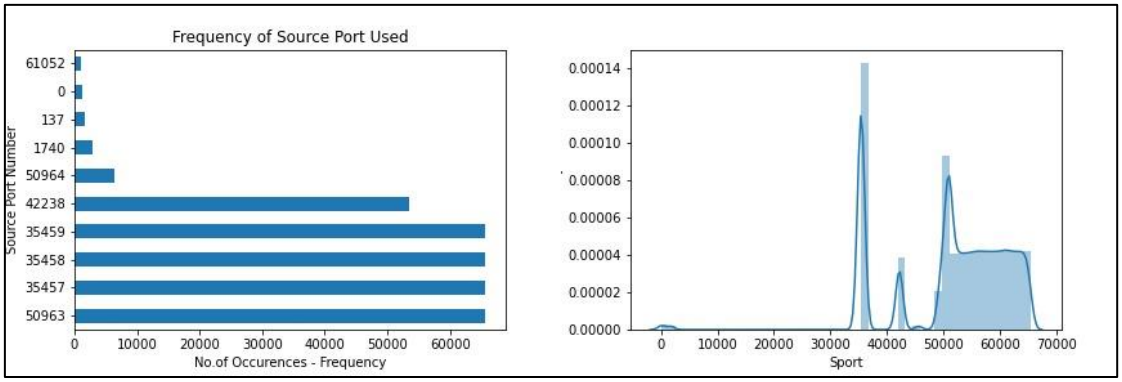

Figure 2. Frequency of Source port used in the dataset with its KDE. Several dynamic port ranges are used while capturing the traffic which ranges in between 49152-65535. 50963 is the most used port in the dataset.

It has been observed that the dataset is imbalanced in nature, it happens when observation of one class is higher or lower compare to the other class.

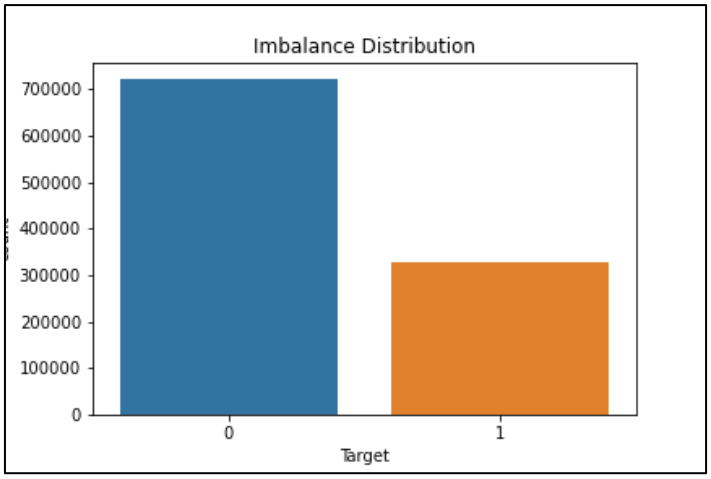

Figure 3. Dataset imbalance data distribution

Large number of majority class (target 0 – normal traffic) have been used, compare to minority class (target 1 – attack traffic).

The dataset has a greater number of major points (positive class), compare to minor points (negative class) [51].As a result, learning algorithm will be more bias towards positive class and tends to ignore the negative class, this will increase the prediction rate of majority class and misclassify the minority class. This imbalance will not make proper model for learning, and hence no true result. This imbalance distribution in the target class can be seen in the Figure3, where target 0 (normal traffic) is higher than target 1 (attack traffic). 68% of data consist of normal traffic and 31% of data is of attack traffic.We can resolve this imbalance constraint using sampling technique [52], where we can balance the data of both class, which can then further send to learning process. NearMiss,SMOTE [53], RandomUnderSample are some of the algorithms which work on the basis of undersampling and oversampling, to have a balanced dataset for both classes [54] have proposed an undersampling algorithm – Membership Probability–based undersampling (MPU), a low computational complexity algorithm which can balanced the dataset according to the membership probability to the majority class. In addition to that, it will decrease the impact of data loss, since it removes fewer data and give good performance.

NearMiss [55] – an undersampling algorithm which also aims in balancing imbalance class distribution by randomly removing majority class. The idea is, when instances of two different classes are very near to each other, nearmiss() will eliminate the instance of majority class to create the space between two classes, which will help in classification process. near_neighbours options of nearmiss() can prevent the data loss by calculating the average distance to minority class. RandomUnderSampler() is another undersampling algorithm, which will randomly selecting the data from the majority class and removing them from the training set.This process can be repeatedly done, until the desired class distribution is not achieved. On contrary, we have RanomOverSampler() – which will randomly selecting the data from minority class, with replacement and add it to the training set. This will duplicate the examples in the dataset, but it is an efficient method when the dataset is affected by skewed distribution.[56] Synthetic Minority Oversampling technique (SMOTE) is an oversampling technique – which will randomly increase minority class by replicating them. First it will set the minority class of set A, where each $x \in A$,for each x, k-nearest neighbor can be obtained using Euclidean distance between x and every other sample of minority class. A_1will be constructed by randomly selecting samples from the k-nearest neighbor set. New training samples (X`) can be generated using $x` = x + rand(0,1) * | x-x_k |$, where $x_k \in A_1$. SMOTE uses linear interpolation to generate these new imaginary samples, which will balance the dataset. We have applied undersampling and oversampling to our dataset, however it is not recommended to use oversampling as it is naïve nature. Researchers have improved oversampling technique using SMOTE-like techniques, adopted in our dataset as well shown in Table 4**.**There is not much work done on time complexity of such algorithms, hence it is difficult to know which oversampling or undersampling method should be considered. Although SMOTE techniques are advisable as it gives efficient result in classification. BorderlineSMOTE and SMOTETomek takes too much processing time to sample the large dataset, therefore in this paper, we have used normal SMOTE and balance the class distribution.

Technique	X Shape	Y Shape	Target 0 – Normal Traffic	Target 1 – Attack Traffic
Original Data	1048575	1048575	721827	326748
NearMiss	653496	653496	326748	326748
RandomUnderSample	653496	653496	326748	326748
RandomOverSample	1443654	1443654	721827	721827
SMOTE	1443654	1443654	721827	721827
BorderlineSMOTE	1443654	1443654	721827	721827
SMOTETomek	1443654	1443654	721827	721827

Table 4: Resample of Dataset using various resampling techniques – Undersample and Oversample

V. Result and Analysis

To give an overview of performance analysis of different classification algorithm on this dataset, we have applied ensemble method, logistic classifier models, naïve bayes, tree classifiers and discriminant analysis on this dataset to evaluate its performance.

Learning Algorithm	Train Accuracy	Test Accuracy	Precision	Recall	AUC	F1-Score	MCC
AdaBoost Classifier	1.0	1.0	0.999	0.999	0.999	0.999	0.99
Bagging Classifier	1.0	1.0	1.0	0.999	0.999	0.999	1.0
ExtraTrees Classifier	1.0	1.0	1.0	1.0	1.0	1.0	1.0
GradientBoosting Classifier	1.0	1.0	0.999	0.999	0.999	0.999	0.99
RandomForest Classifier	1.0	1.0	0.999	1.0	0.999	0.999	0.99
Decision Tree Classifier	1.0	1.0	0.999	1.0	0.999	0.999	1.0
Logistic Regression CV	0.998	0.998	0.998	0.998	0.998	0.998	0.99
Perceptron	0.998	0.998	0.998	0.998	0.998	0.998	0.99
PassiveAggresive Classifier	0.998	0.998	0.998	0.998	0.998	0.998	0.99
SGD Classifier	0.998	0.998	0.998	0.998	0.998	0.998	0.99
Quadratic Discriminant Analysis	0.997	0.997	0.997	0.998	0.997	0.997	0.99
Guassian NB	0.996	0.996	0.994	0.998	0.996	0.996	0.99
Linear Discriminant Analysis	0.996	0.996	0.993	0.998	0.996	0.996	0.99
BernoulliNB	0.812	0.811	0.990	0.629	0.811	0.769	0.67
Ridge Classifier CV	0.396	0.396	0.347	0.233	0.397	0.279	0.03

Table 5: Comparison of different classifiers against various performance parameters of classification.

Result shows us that ensemble learning is comparably giving better performance than others, this result is more fine-tuned compared to the result without sampling the dataset.

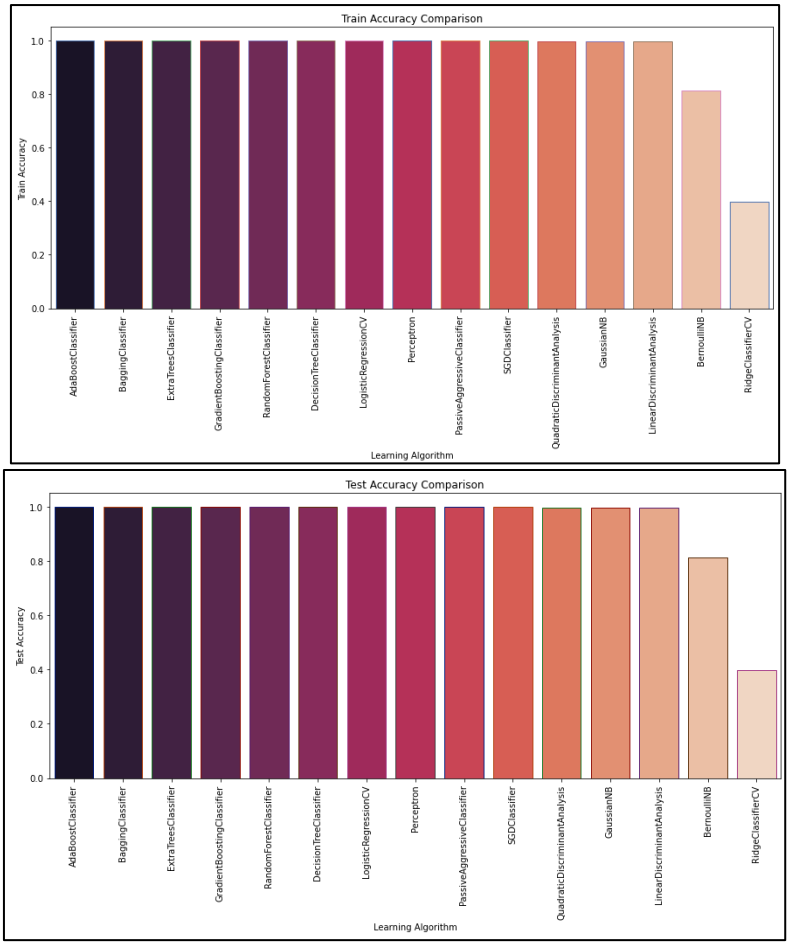

Figure :4. Representation of various classifiers with its train and test accuracy after balanced data distribution.

There is a high-risk factor in application such as fraud detection, anomaly detection and intrusion detection, if we work with imbalance dataset, as the result will always bias towards majority class and hence any negative class can be interpreted as positive class by the administrator. The comparison of different algorithm with balanced dataset (prepared by SMOTE) is shown in Table 4. Moreover, the accuracy comparison of train and test data is also shown in Figure 4**.** Accuracy is not always a coherent performance metric for learning models, it is necessary to check our models against other metric such as confusion matrix and its supplementary measurement - precision, recall and F-measure. Confusion matrix [57] has several parameters, on the basis of those parameters other measurement can be done, shown in Table 6.

Parameters	Definition
True Positive Rate (TPR)	Attack occurs and alarm raised
False Positive Rate (FPR) – Type 1 Error	No Attack, but alarm raised
True Negative Rate (TNR)	No Attack and no alarm raised
False Negative Rate (FNR)– Type 2 Error	Attack occurs, but not alarm raised

Table 6.Confusion Matrix in the context of Intrusion detection system

Accuracy – It is the ratio of all True positive and True Negative to the total samples. Higher the value of accuracy, better the learning model. However, accuracy is not a good measure in classification problem, it rarely considers the FP and FN while calculating the accuracy, as FN and FP are errors it should decrease the accuracy, but it never happens.

$$\text{Accuracy} = \frac{TP+TN}{Total}$$

Precision – It is the ratio of True positive to the total true positive and false positive samples. Out of all positive class, how many are actually positive, we have predicted correctly is precision going to indicate. Higher the better**.**

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

Recall – It is the ratio of True Positive to the total true positive and false negative sample. Out of all positive class, how much we have predicted correctly specify by the recall. Higher the better.

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

Precision and Recall helps in comparing different algorithms, fromTable 5,it can be seen ExtraTree classifier has given a better performance, their precision and recall are far better than others. Another measure is F-β measure or F1-score which will identify which learning model should be used, when each system has better precision and recall.

$$\text{F1 Score} = \frac{1}{\beta * \frac{1}{Precision} + (1-\beta) * \frac{1}{Recall}}$$

It depends on β value, greater β, greater importance to precision and lower β, higher recall. Receiver Operator Characteristic (ROC) is a measurement, where if we exactly want to know that an attack is occurred or not, we can set a threshold between 0 and 1. ROC can be represented between true positive rate (sensitivity) and false positive rate (1-specificity), the roc value coming to optimal threshold (near to 0) is better, it is shown in Figure 5.

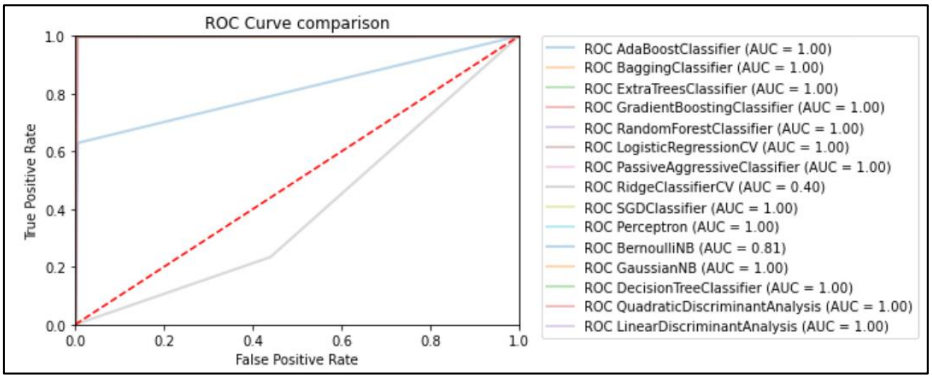

Figure 5. Representation of ROC curve for different classifiers, ensemble classifiers achieve optimal threshold and has good AUC score.

Performance of ROC with our dataset is satisfactory, where AdaBoost, Bagging and ExtraTree classifier comes in optimal threshold point with AUC of 1.0. The result also shows us sensitivity (true positive rate) – which says % of attack correctly identified as attack. The value of true positive rate is high, compare to false positive rate, indicates we should use ensemble techniques with any IDS/IPS to detect normal and attack traffic. Matthews Correlation Coefficient (MCC) a.k.a "phi coefficient" used to classify binary classification problem; it is closely related to chi-square. In terms of confusion matrix, it is defined as,

$$\text{MCC} = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

VI. Future Work

Due to severity of SCADA testbed our future work is to create an effective simulation of control system wherein we can perform several network attacks and hence a concrete dataset can be generated. Upon which can use deep learning – alternative to ML, which can categorize network traffic in more efficient way. Another possible area is to integrate such classifiers in our detection system so that it can detect the anonymous or attack traffic is less time.

VII. Conclusion

In any control system such SCADA, it is imperative to detect an external or internal intrusion. In this paper, we have discussed SCADA system security challenges which can manipulate small subsystem such PLC to big systems such as fire safety system or historian. In addition to that, communication between different components of SCADA needs communication protocols which has wide drawbacks and can be easily manoeuvred by the attacker. IDS/IPS has a great viability to detect any unknown or known traffic, to increase capability of such

detection system, we have incorporated machine learning algorithms. To observe the feasibility and reliability of these algorithms we have evaluated and compared for better prediction of intrusion on SCADA dataset. Since the dataset we have used was imbalanced, has more positive classes compared to negative classes. SMOTE algorithms are used for resampling the data that gives equal data distribution of both classes. Next, we have applied 11 classifiers and compared with different performance metrics. It has been evaluated, that ensemble classifiers are superior in detecting the known and unknown traffic. As, accuracy is not a metric in classification problem, we have used precision, recall F1-score, MCC and ROC curve to justify ensemble method as a best classifier.

VIII. References

[1] Parian C, Guldimann T, Bhatia S (2020) Fooling the Master: Exploiting Weaknesses in the Modbus Protocol. Procedia Computer Science 171:2453-2458.

[2] "21 Steps to Improve Cyber Security of SCADA Networks", US Department of Energy, 2003.

[3] L. Rosa, M. Freitas, S. Mazo, E. Monteiro, T. Cruz, and P. Simões, "A comprehensive security analysis of a SCADA protocol: From OSINT to mitigation," IEEE Access, vol. 7, pp. 42156–42168, 2019.

[4] Li, Dong, Huaqun Guo, Jianying Zhou, Luying Zhou, and Jun Wen Wong, "SCADAWall: A CPI-enabled firewall model for SCADA security" Computers & Security 80: 134–154. 2019.

[5] Rezai, P. Keshavarzi, and Z. Moravej, ''Key management issue in SCADA networks: A review,'' Eng. Sci. Technol., Int. J., vol. 20, no. 1, pp. 354–363, 2017.

[6] Modbus Organization Inc., Modbus Application Protocol Specificationv1.1b3, 2012.

[7] G. Clarke, D. Reynders and E. Wright, "Differences between DNP3 and IEC 60870", Practical Modern SCADA Protocols, pp. 307-311, 2003.

[8] G. De La Torre Parra, P. Rad and K. Choo, "Implementation of deep packet inspection in smart grids and industrial Internet of Things: Challenges and opportunities", Journal of Network and Computer Applications, vol. 135, pp. 32-46, 2019.

[9] Dutta N., Jadav N., Dutiya N., Joshi D., "Using Honeypots for ICS Threats Evaluation". In: Pricop E., Fattahi J., Dutta N., Ibrahim M. (eds) Recent Developments on Industrial Control Systems Resilience. Studies in Systems, Decision and Control, vol 255. Springer, Cham. 2020.

[10] E. Knapp, "Industrial Network Protocols", Industrial Network Security, pp. 55-87, 2011.

[11] M.Mallouhi, Y. Al-Nashif, D. Cox, T. Chadaga, S. Hariri "A test bed for analyzing security of SCADA control systems (TASSCS)," IEEE PES Innovative Smart Grid Technologies (ISGT), pp. 1-7. Jan 2011.

[12] M. Erol-Kantarci, and H. T. Mouftah, "Smart grid forensic science: applications, challenges, and open issues," IEEE Communications magazine, vol. 51,no. 1, pp. 68-74, Jan 2013.

[13] S. Ghosh and S. Sampalli, "A Survey of Security in SCADA Networks: Current Issues and Future Challenges," in IEEE Access, vol. 7, pp. 135812-135831, 2019.

[14] R. E. Johnson, "Survey of SCADA security challenges and potentialattack vectors," in Proc. IEEE Int. Conf. Internet Technol. SecuredTrans. (ICITST),pp. 1–5, London, U.K., 2010.

[15] E. E. Miciolino, G. Bernieri, F. Pascucci and R. Setola, "Communications network analysis in a SCADA system testbed under cyber-attacks," 2015 23rd Telecommunications Forum Telfor (TELFOR), pp. 341-344, 2015.

[16] E. Etcheves Miciolino´ et al., "FACIES: A Testbed for Distributed Faultand Attack Identification in Interdependent Critical Infrastructures," in2nd International SCADALab Workshop, Seville (Spain), 2014.

[17] IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) Framework and Rules, 1516.2-2010.

[18] Sajid Nazir, Shushma Patel, Dilip Patel, Assessing and Augmenting SCADA Cyber Security-A Survey of Techniques, Computers & Security (2017).

[19] A. Choubineh, D. Wood and Z. Choubineh, "Applying separately cost-sensitive learning and Fisher's discriminant analysis to address the class imbalance problem: A case study involving a virtual gas pipeline SCADA system", International Journal of Critical Infrastructure Protection, vol. 29, p. 100357, 2020.

[20] T.H. Morris, Z. Thornton, I. Turnipseed, Industrial control system simulationand data logging for intrusion detection system research, in: Proceedings of the 7th Annual Southeastern Cyber Security Summit, 2015.

[21] T. Morris, W. Gao, Industrial control system traffic data sets for intrusion detection research, in: Proceedings of the International Conference on Critical Infrastructure Protection, Springer, Berlin, Heidelberg, pp. 65–78, 2014.

[22] K. O. Akpinar and I. Ozcelik, "Analysis of Machine Learning Methods in EtherCAT-Based Anomaly Detection," in IEEE Access, vol. 7, pp. 184365-184374, 2019.

[23] S. V. B. Rakas, M. D. Stojanović and J. D. Marković-Petrović, "A Review of Research Work on Network-Based SCADA Intrusion Detection Systems," in IEEE Access, vol. 8, pp. 93083-93108, 2020.

[24] S. Nazir, S. Patel, and D. Patel, ‘‘Assessing and augmenting SCADA cybersecurity: A survey of techniques,’’ Comput. Secur., vol. 70, pp. 436–454,Sep. 2017.
[25] Y. Geng, Y. Wang, W. Liu, Q. Wei, K. Liu, and H. Wu, ‘‘A survey of industrial control system testbeds,’’ IOP Conf. Ser. Mater. Sci. Eng., vol. 569, no. 4, Art. no. 042030, 2019.
[26] M. M. S. Khan, A. Palomino, J. Brugman, J. Giraldo, S. K. Kasera and M. Parvania, "The Cyberphysical Power System Resilience Testbed: Architecture and Applications," in Computer, vol. 53, no. 5, pp. 44-54, May 2020.
[27] H. Gao, Y. Peng, K. Jia, Z. Dai and T. Wang, “The Design of ICS Testbed Based on Emulation, Physical, and Simulation (EPS-ICS Testbed),” 2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 420-423, 2013.
[28] A. A. Farooqui, S. S. H. Zaidi, A. Y. Memon and S. Qazi, “Cyber Security Backdrop: A SCADA testbed,” 2014 IEEE Computers, Communications and IT Applications Conference, pp. 98-103, 2014.
[29] A. Almalawi, Z. Tari, I. Khalil and A. Fahad, “SCADAVT-A framework for SCADA security testbed based on virtualization technology,” 38th Annual IEEE Conference on Local Computer Networks, pp. 639-646, 2013.
[30] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah and F. Ahmad, “Network intrusion detection system: A systematic study of machine learning and deep learning approaches”, Transactions on Emerging Telecommunications Technologies, vol. 32, no. 1, 2020.
[31] M. Zolanvari, M. A. Teixeira, L. Gupta, K. M. Khan and R. Jain, “Machine Learning-Based Network Vulnerability Analysis of Industrial Internet of Things,” in IEEE Internet of Things Journal, vol. 6, no. 4, pp. 6822-6834, Aug. 2019.
[32] Yang H, Liu X, Zhang D et al. (2021) Machine learning for power system protection and control. The Electricity Journal 34:106881. 2021.
[33] S. V. B. Rakas, M. D. Stojanovic, and J. D. Markovic-Petrovic, ‘‘A review of research work on network-based SCADA intrusion detection systems,’’ IEEE Access, vol. 8, pp. 93083–93108, May 2020.
[34] R. Atallah and A. Al-Mousa, “Heart Disease Detection Using Machine Learning Majority Voting Ensemble Method,” 2019 2nd International Conference on new Trends in Computing Sciences (ICTCS), pp. 1-6, 2019.
[35] Schapire R.E. “The Boosting Approach to Machine Learning: An Overview”. In: Denison D.D., Hansen M.H., Holmes C.C., Mallick B., Yu B. (eds) Nonlinear Estimation and Classification. Lecture Notes in Statistics, vol 171. Springer, New York, NY. 2003.
[36] Y. Freund and R. Schapire, “A short introduction to boosting,” J. Jpn. Soc. Artif. Intell., vol. 14, no. 5, pp. 771–780, Sep. 1999.
[37] Q. Sun and B. Pfahringer, ‘‘Bagging ensemble selection,’’ in Advances in Artificial Intelligence (Lecture Notes in Artificial Intelligence), vol. 7106, D. Wang and M. Reynolds, Eds. Berlin, Germany: Springer-Verlag, pp. 251–260, 2011.
[38] E. Yaman and A. Subasi, “Comparison of Bagging and Boosting Ensemble Machine Learning Methods for Automated EMG Signal Classification”, BioMed Research International, vol. 2019, pp. 1-13, 2019.
[39] K. Machova, M. Puszta, F. Barcak and P. Bednar, “A comparison of the bagging and the boosting methods using the decision trees classifiers”, Computer Science and Information Systems, vol. 3, no. 2, pp. 57-72, 2006.
[40] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001).
[41] Y. Zhao, “Decision Trees and Random Forest”, R and Data Mining, pp. 27-40, 2013.
[42] C. Peng, K. Lee and G. Ingersoll, “An Introduction to Logistic Regression Analysis and Reporting”, The Journal of Educational Research, vol. 96, no. 1, pp. 3-14, 2002.
[43] Y. Huang and L. Li, “Naive Bayes classification algorithm based on small sample set”, 2011 IEEE International Conference on Cloud Computing and Intelligence Systems, 2011.
[44] “Bayes' theorem – Wikipedia”, en.wikipedia.org, 2021.
[45] G. Singh, B. Kumar, L. Gaur and A. Tyagi, ‘Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification,” 2019 International Conference on Automation, Computational and Technology Management (ICACTM), pp. 593-596, 2019.
[46] A. Tharwat, T. Gaber, A. Ibrahim and A. Hassanien, “Linear discriminant analysis: A detailed tutorial”, AI Communications, vol. 30, no. 2, pp. 169-190, 2017.
[47] J. Shlens.,“A tutorial on principal component analysis”, (2005, December).
[48] Gumaei A, Hassan M, Huda S et al., “A robust cyberattack detection approach using optimal features of SCADA power systems in smart grids”, Applied Soft Computing 96:106658. 2020.
[49] A. Wosiak and D. Zakrzewska, “Integrating Correlation-Based Feature Selection and Clustering for Improved Cardiovascular Disease Diagnosis”, Complexity, vol. 2018, pp. 1-11, 2018.
[50] Teixeira M (2021) WUSTL-IIOT-2018 Dataset for ICS (SCADA) Cybersecurity Research. In: Cse.wustl.edu. https://www.cse.wustl.edu/~jain/iiot/index.html.
[51] D. Veganzones and E. Séverin, “An investigation of bankruptcy prediction in imbalanced datasets”, Decision Support Systems, vol. 112, pp. 111-124, 2018.

[52] A. More, “Survey of resampling techniques for improving classification performance in unbalanced datasets” in arXiv:1608.06048, Aug. 2016.
[53] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer,“SMOTE: Synthetic Minority Over-Sampling Technique,”J. Artificial Intelligence Research, vol. 16, pp. 321-357, 2002.
[54] Ahn, G., Park, YJ. & Hur, S. A Membership Probability–Based Undersampling Algorithm for Imbalanced Data. J Classif. 2020.
[55] N. Gurevich, S. Markovitch and E. Rivlin, “Active Learning with Near Misses”, American Association for Artificial Intelligence, 2006.
[56] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16:321–357, 2002.
[57] Ting K.M, “Confusion Matrix.” In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning and Data Mining. Springer, Boston, MA, 2017.