IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 September 2023, accepted 13 October 2023, date of publication 23 October 2023, date of current version 30 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3326751

RESEARCH ARTICLE

Validation of a Machine Learning-Based IDS Design Framework Using ORNL Datasets for Power System With SCADA

MARZIA ZAMAN[1], DARSHANA UPADHYAY[1], AND CHUNG-HORNG LUNG[2], (Senior Member, IEEE)

[1]Research and Development Department, Cistel Technology Inc., Ottawa, ON K2E 7V7, Canada
[2]Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada

Corresponding author: Darshana Upadhyay (dupadhyay@cistel.com)

The authors gratefully acknowledge the support for this research by the Natural Sciences and Engineering Research Council (NSERC), Canada through a Discovery Grant.

ABSTRACT Supervisory Control and Data Acquisition (SCADA) systems are widely used for remote monitoring and control of industrial processes, such as oil and gas production, power generation, transmission and distribution, and water treatment. Despite the enhanced accessibility, control, and data availability afforded by recent advances in communication technologies, the utilization of these technologies exposes critical infrastructures such as power systems to potential cyber threats. A Machine Learning (ML)-based Intrusion Detection System (IDS) seems promising; however, the development of ML models often requires custom methodologies for data preprocessing and training. This strategic approach is necessary for creating high-performance models that can be robustly evaluated and seamlessly integrated into real-time systems. As a result, we propose an ML-based IDS design framework for a SCADA-based power system incorporating effective modeling aspects, such as dataset preprocessing to ensure accurate representation, data augmentation for achieving a balanced dataset, automated feature selection to reduce dimensionality, and rigorous model training and testing procedures. To substantiate our proposed design framework, we conducted a series of experiments using a publicly available ORNL (Oak Ridge National Laboratory) dataset for a SCADA-based power system. The evaluation process encompasses efficient validation techniques with unseen data. Furthermore, the augmented dataset emerged through the aggregation of readings from four Phasor Measurement Units (PMUs) collected over a specific time span into a unified dataset. Among the assessed classifiers, the Random Forest (RF) model, trained on an augmented and balanced dataset, outperformed others, yielding an F1 score of 94.09% during testing with unseen data.

INDEX TERMS Intrusion detection, machine learning, generative adversarial network, SCADA systems, cyber-attacks, industrial control systems.

I. INTRODUCTION

Intrusion Detection Systems (IDSs) are critical for modern information security infrastructure. It plays an important role in identifying and alerting administrators about unauthorized access, misuse, or tampering with computer systems and networks. In the case of Supervisory Control and Data Acquisition (SCADA) systems, IDSs are even more critical, given the sensitive nature of the data and systems involved [1].

SCADA systems play a vital role to control and monitor the processes of industrial control systems, such as controlling the production lines in manufacturing units, managing power plants in energy sectors, controlling water distribution, and filtering in water treatment plants. They are large and complex systems that collect and transmit large amounts of real-time data from remote locations to a central control center. The data includes readings from sensors, valves, and other devices, as well as commands and control signals sent to the field devices.

Figure 1 shows the block diagram of a typical SCADA architecture for power systems. As illustrated in Figure 1,

The associate editor coordinating the review of this manuscript and approving it for publication was Pedro R. M. Inácio.

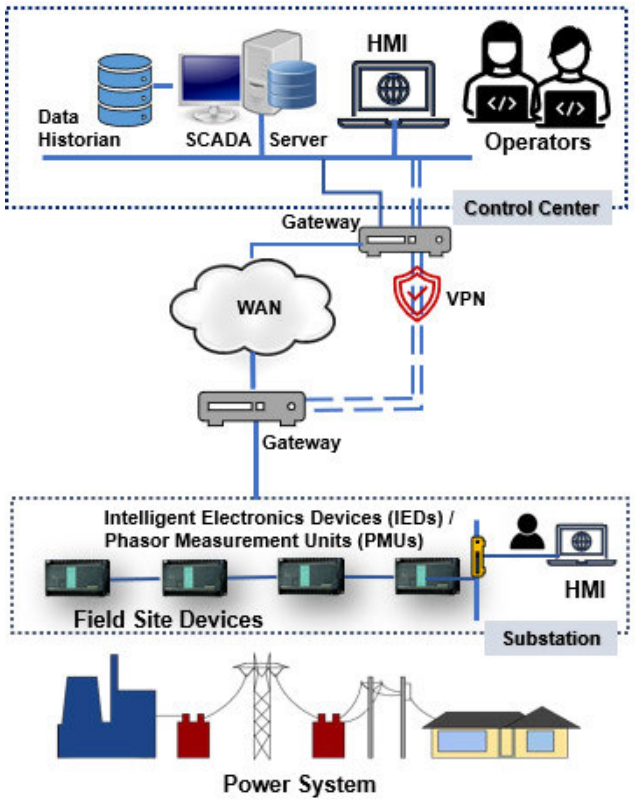

FIGURE 1. Architecture of SCADA-based power systems.

sensors, meters, and PMUs (Phaser Measurement Units) are used at various locations at the substation of the power transmission system to measure diverse electrical data, such as voltage, current, frequency, power factor, and phase angles. Traditionally, field controllers, such as Programmable Logic Controllers (PLCs), Intelligent Electronic Devices (IEDs), or Remote Terminal Units (RTUs) are connected with these measurement units to transmit the data to the SCADA server. The digital data received from these field devices are then transmitted to the Master Terminal Unit (MTU) located at a remote location. Remote monitoring of the processes within the power system is viable through dedicated secure Virtual Private Network (VPN) lines or by utilizing Wide Area Network (WAN) to establish point-to-point networks. However, control and parameter changes may be carried out either locally at power substations or remotely, by operators within designated control rooms. The MTU determines the set points based on parameter ranges and sends signals back to the field site components for necessary actions to avoid malfunctions and optimize the performance of the system.

While many aspects of modern power systems are automated to enhance reliability and efficiency, human intervention remains crucial for several purposes such as operating and controlling breakers or manually disconnecting a line. The Human-Machine Interface (HMI) serves as the bridge between human operators and the plant floor machinery, ensuring the smooth and effective flow of electric energy. This includes configuring critical parameters of the power transmission systems to ensure they operate within safe and optimal limits. Moreover, unexpected events or emergencies may necessitate rapid actions by human operators. For instance, in the case of a sudden power surge or a system fault, operators can use the HMI to isolate the affected area, reroute power, or initiate safety protocols. Furthermore, system monitoring involves taking corrective actions in response to alarms and alerts. Maintenance or repairs require operators in the control room to coordinate activities effectively.

Recent advancements, such as the integration of new data sources like IoT, Industry 4.0, and phasor measurement units, increase the risk of cyberattacks reaching SCADA systems in power grids. While the power system SCADA is not generally connected to the internet, it is crucial to recognize the potential vulnerabilities within these systems. In addition to insider attacks, there are various other types of cyberattacks that pose a significant threat to power grid SCADA systems. Some of these include malware infections through USB, Denial-of-Service (DoS) Attacks, Zero-Day Exploits owing to vulnerable SCADA components, and Data Manipulation attacks.

Real-time cyber attacks can have devastating consequences, including, but not limited to, service disruption as observed in 2003 in the Davis-Besses nuclear power plant [2] and in 2008 at the Hatch nuclear power plant in Baxley, Georgia [2] when these stations were attacked by cybercriminals. Recent cyber-attacks in industrial control systems include a ransomware attack targeted at Colonial Pipeline Inc. in the US in May 2021 [3] bringing the facility to a complete halt for a few days. The Ukraine power plant cyber attack reported in 2015 [4] was probably the first known successful attack on power grids where attackers were able to disrupt the electricity supply to the end users. Power grid attack is one of the most critical issues in industrial control systems and it is essential to protect them by applying adequate security measures [5].

Intrusion detection in SCADA systems is challenging due to the unique requirements and constraints of these systems. These requirements include (i) real-time response, (ii) scalability, (iii) interoperability, and (iv) reliability. SCADA systems require real-time monitoring and response to events. The Intrusion Detection System (IDS) must be able to process large amounts of data in real time and alert administrators quickly in case of an intrusion. SCADA systems often have a large number of devices and sensors. Therefore, the IDS must be able to handle this scale and still provide fast and accurate intrusion detection. In addition, these systems often involve multiple different protocols, devices, and systems from various vendors. As a result, the IDS must be able to operate seamlessly in such heterogeneous environments. Further, SCADA systems are often mission-critical, and therefore, their operations cannot be disrupted.

Thus, IDS must be reliable, with a low false positive rate, and must not cause any disruption to the normal operation of the system.

SCADA systems also include resource constraint devices, such as Remote Terminal Unit (RTU) and field devices. Therefore, all related security protocols and algorithms must be able to run on resource-constrained devices if a host IDS is considered. IDSs can be performed using two main approaches: signature-based and anomaly-based. Signature-based intrusion detection systems rely on predefined patterns of behavior or signatures to identify intrusions. These signatures are based on knowledge of past intrusions and the behavior of known attackers. The advantage of this approach is its accuracy, but it can be vulnerable to new and unknown intrusions. On the other hand, anomaly-based intrusion detection systems use machine learning algorithms to identify unusual or abnormal behaviors within the SCADA system. This approach is more robust against new and unknown intrusions, but it has a higher false positive rate. To mitigate this, anomaly-based IDS often employs a two-stage approach, where the first stage identifies potential intrusions and the second stage confirms or denies the intrusion using additional information.

In this research work, our objective is to develop the ML-based IDS framework, more specifically an Offline Training Module (OTM) for SCADA-based power systems. We work towards developing a generalized framework that allows us to develop a robust ML-based IDS model that can perform with high accuracy when tested with new or unseen data. This framework enables us to pre-process the raw dataset, augment and balance dataset that maybe imbalanced, select the most useful features, save the best model during cross-validation and validate the model further with unseen data. We created synthetic data using Generative Adversarial Network (GAN) to augment the dataset such that we obtain more training data as well as balance the dataset. GAN models have shown to be quite effective in a wide range of machine learning applications, including tabular data generation [6]. Data augmentation increases the generalizability of a data model and thereby minimizes overfitting. We used a publicly available power system SCADA dataset, collected by a group of researchers at Mississippi State University and Oak Ridge National Laboratory (ORNL) [7], hereafter referred to as the ORNL dataset.

A. MAJOR CONTRIBUTIONS

The main contributions of this work are the following.

- We propose a robust ML-based IDS design framework for SCADA-based power systems with two main components: the Offline Training Module (OTM) and the Online Detection Module (ODM). These modules are further validated using several testing approaches by aggregating 15 small datasets from four locations into one to create a high-performance model within a distributed environment.
- We incorporate effective modeling aspects such as data augmentation and feature selection methodology to develop a balanced dataset for improved performance during model training.
- We conducted a series of experiments encompassing effective validation techniques to evaluate and compare three tree-based classical machine learning (ML) algorithms. Instead of performing only on cross-validation, we tested our best model using unseen data. All models were evaluated in terms of accuracy, precision, recall, F1-Score, False Positive Rate (FPR), and False Negative Rate (FNR) for effective performance measurement.
- We compare the results of our proposed approach with those of published state-of-the-art techniques in terms of accuracy, features, and size of the dataset.

The rest of the paper is organized as follows. In Section II, we present the background related to the architecture of SCADA-based power systems along with a literature survey and research scope by referring to published IDS techniques. In Section III, we focus on a proposed framework for IDS for power systems. Section IV briefly describes the power system datasets used in this study. The synthetic data generation and feature selection module is covered in Section V. Model design and development, including the methodology used to preprocess, augment, and analyze the datasets, as well as model training and validation, are discussed in Section VI. Next, in Section VII, we present our results and discuss them through various experiments. Finally, we conclude the paper in Section VIII with final remarks.

II. BACKGROUND AND LITERATURE REVIEW

In recent years, power systems have become increasingly digitized, with many functions automated and controlled remotely. This advancement has led to an increased risk of cyber-attacks on power systems, which could have devastating consequences, such as power outages or even physical damage to the power grid [15]. In order to detect such attacks in power systems, the placement of an efficient Intrusion Detection System is a crucial factor.

Any anomaly in these data may be due to malfunction of the equipment and/or faults in the power system or cyber-attacks modifying the measured values. Cyber-attacks can also result in modifying legitimate commands of the users to control the power system and therefore can have devastating consequences. The collected data are stored in data historians, and log files are maintained. The Human Machine Interface (HMI) component is used to display this information, from where operators and technicians can monitor and control the systems. To ensure the prompt detection of intrusions that occur at these units, the intrusion detection system can be strategically placed, such as on the plant floor and control center.

There are three primary categories of IDS, namely, host-based, network-based, and hybrid IDS. Typically, host-based IDS (HIDS) analyzes activities on individual hosts to detect intrusions, such as local attacks that may not be visible

TABLE 1. An overview of published intrusion detection approaches in power grid systems.

Contributions	Hink et al. 2015 [8]	Pan et al. 2015 [9], [10]	Keshk et al. 2017 [11]	Moustafa 2018 [12]	Keshk et al. 2019 [13]	Upadhyay et al. 2020/2021 [14], [15]
Dataset used	Oak Ridge National Laboratories (ORNL) - power grid dataset [16]					
Feature Selection Method	Not Applied (100%)	Not Applied (100%)	PCC (75%)	ICA (subset of features)	PCC (25%)	GBFS (12%), RFE-GB (25%)
Methodology	Adaboost - JRIP	Common Path Mining	EM Clustering Algo	Beta MHMMs	Gaussian Mixture	Tree-Based, Majority Vote
Approach	Supervised Machine Learning	Data Mining/Pattern matching	Max. Likelihood	Stat. Common	Bayesian Filtering	Machine Learning
Key Features & shortcomings	Low accuracy and low execution speed	Improved attack identification and location. Moderate accuracy and speed	Improved speed but lower accuracy with multi-class datasets	Improved accuracy and location. No multi-attack vectors classification	Improved accuracy, no multi-attack vectors	Improved accuracy and speed but results were based on single location

on the network. However, such IDS is system dependent and typically monitors the logs of individual power grid components. On the other hand, network-based IDS (NIDS) monitors the network traffic of power grid components to detect cyber-attacks. Such IDs typically used Machine Learning models to detect the anomaly and the performance of such IDSs is evaluated using the detection rate and false alarm rate. Hybrid IDS combines the capabilities of network-based and host-based detection to improve the accuracy and reliability of the model. However, hybrid IDS requires more resources for implementation as the dataset used to model such IDS relies on power grid device logs and network traffic.

Several approaches are being adopted by researchers for the development of intrusion detection systems in power systems for accurate intrusion detection. One such approach is signature-based intrusion detection, which uses a pattern-matching technique to determine the signature of malicious events by comparing incoming traffic with stored signatures. However, this technique is only effective for known attacks and is not suitable for real-time systems [8]. Intrusion detection in power grids can also be performed using signal processing techniques such as time-series analysis, wavelet transform, and Fourier analysis. These techniques analyze power grid data in the frequency and time domains to identify anomalies. Such studies have demonstrated the effectiveness of signal processing-based IDSs in SCADA-based power grids. For example, in [17], the authors proposed a wavelet-based IDS for the detection of intrusions into the power grid, which demonstrated high accuracy and low false positive rates. Similarly, in [18], authors developed a Fourier-based IDS for power grid security that achieved high detection rates and low false alarm rates.

As discussed previously, traditional IDS techniques rely on predefined rules and are limited to known attacks; however, AI-based IDS utilizes machine learning algorithms to detect anomalies and identify previously unknown threats, making it more effective. In [19], the authors discuss security threats, vulnerabilities, and cyber-physical system attacks and propose a deep learning-based IDS that analyzes malicious URLs. Reference [20] investigates the vulnerabilities of deep learning-based IDS in power grids to adversarial attacks, proposing defense mechanisms to enhance robustness. Additionally, [21] examines the effectiveness of gradient-boosting algorithms for anomaly detection in imbalanced data within power grids. Furthermore, [22] presents a scalable anomaly detection engine for large-scale smart grids, capable of distinguishing actual faults from disturbances and intelligent cyber-attacks.

With an increasing number of cyber-attacks targeting power systems, the need for intrusion detection models for SCADA systems is becoming more critical. According to a report by the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT), there was a 20% increase in the number of cyber incidents targeting critical infrastructure in 2020 [23]. Hence, the proper design and development of an intrusion detection model is one of the crucial factors to secure such critical infrastructure.

Researchers have introduced a range of intrusion detection techniques aimed at bolstering the security of SCADA-based power grids. Table 1 offers an overview of the existing literature on intrusion detection systems (IDSs) specifically designed for power grids. To ensure a fair comparison, we have compiled summaries of published techniques for robust IDSs that exclusively utilize ORNL datasets. In a study by Hink et al. [8], they conducted a comparative analysis of various machine learning techniques using a power grid dataset and singled out Adaboost-JRIP as a highly effective classifier. However, it is worth noting that their study did not incorporate dataset dimensionality reduction, leading to less-than-optimal accuracy and slower execution speeds.

Another line of research by Pan et al. focused on hybrid intrusion detection systems (IDS) that leverage data mining techniques. They specifically employed common path mining to identify attack locations in their work [9], [10]. In a separate study [11], the authors adopted the Pearson Correlation Coefficient (PCC) for feature selection, retaining 75% of the features, and employed the Expectation Maximization

Clustering Technique (EMCT) for event classification. While this approach succeeded in enhancing execution speed, it did not bring about significant improvements in accuracy, especially for multi-class datasets.

Moustafa et al. took a different approach by using Independent Component Analysis (ICA) for feature selection and the Beta Mixture Hidden Markov (BMHM) classification model, which yielded promising accuracy results in their research [12]. However, their study concentrated on a subset of the features, and therefore, the exact number of features used remains unspecified. Furthermore, this technique was further refined by combining PCC with the Gaussian Mixture - Kalman Filter Model (GMM-KF) in another study [13], where the authors achieved a reduction in feature usage to 25%, resulting in improved accuracy and execution speed. However, it is essential to note that this experiment was limited to a binary dataset.

Upadhyay et al. [14] proposed Gradient Boosting Feature Selection (GBFS) technique to extract important features before applying the classifiers. The researchers performed cross-validation over 15 ORNL power system datasets with the top 15 features selected using GBFS. The same research group [15] has also proposed an integrated SCADA IDS framework for power systems where they used the Recursive Feature Elimination technique to select the top 30 features and majority voting using nine heterogeneous classifiers. Their proposed methodology when applied to ORNL power system datasets yields high accuracy for binary classification. However, the results demonstrated in that study were based on a single data file out of the fifteen data files rather than aggregating all 15 files into one dataset for training and validation.

Current research on SCADA-based power systems has identified several gaps in the recent literature. One of the challenges in developing effective intrusion detection models for SCADA-based power systems is the lack of publicly available datasets that can be used to train and test these models. Some researchers have called for the development of standardized datasets for SCADA intrusion detection research [24]. This research gap opens the need for generating a synthetic dataset from the existing dataset for effective use and better validation of ML models. While there has been prior research on employing machine learning techniques for SCADA intrusion detection, there remains a demand for more extensive investigations that delve into the efficacy of a generic IDS model that includes feature selections, data augmentation, and effective ML validation techniques [25].

Intrusion detection models that generate a large number of false positives can negatively impact the performance of SCADA-based power systems. However, there is a lack of studies that address the issue of false alarms. There has been limited research on optimizing intrusion detection models to minimize false positives while still detecting real intrusions [26]. Furthermore, there is a lack of a generalized model that helps detect cyber-attacks while building a centralized IDS model for power systems. Additionally, there is a lack of significant studies that specifically address the validation of test models using an effective approach to assess the efficiency of the model.

III. PROPOSED FRAMEWORK

To address the research gaps mentioned in the previous section, we propose a generalized and robust ML-based IDS framework for SCADA-based power systems. Adopting this generic framework can enhance the reliability and flexibility of the system, as it allows us to develop a better IDS model and continuously improve it. Using this framework, we can conduct data aggregation and data pre-processing including normalization, data balancing and augmentation, feature selection, algorithm selection as new data becomes available and holdout validation of the resulting model using unseen test data. Integrating various ML techniques to develop, continuously train, improve, and properly validate the ML-based IDS model using a reasonably large training dataset with real and synthetic data could lead to a more effective and robust intrusion detection system.

In this section, we have provided a detailed description of our vision for machine learning (ML)-based IDS for power systems. Figure 2 represents the main building blocks of the ML-based IDS, which consists of two main software modules, namely, the Offline Training Module (OTM) and the On-Line Detection Module (ODM). The OTM is responsible for training, evaluating, selecting, and sharing the best model to the ODM. On the other hand, the ODM implements the intrusion detection pipeline in the MTU and/or RTU and detects the intrusion in real-time. The ODM detects the intrusion based on the model developed in the ML server. The ODM is also responsible for providing the SCADA data to the OTM so that it can continue to update the model based on new data. The collected data are labeled and stored in the database. In OTM, a sliding window can be used to select the most recent dataset for training as the IDS continues to collect new data. The normalization scale, selected features, and best model are shared periodically with the ODM by the OTM to ensure these data including the best model are up to date. The normalization scale is used while processing the data in RTU and MTU prior to using the model for intrusion detection. Data processing in the OTM includes data cleaning, imputation, normalization, and augmentation, while the imputation and augmentation components are not present in the ODM.

Our goal is to develop and validate a novel ML-based IDS Design framework that can be easily integrated into various places in power systems. Aggregating data over a longer period (e.g., 1 data from 1 file vs. 15 files) and from multiple PMUs can be more effective as the volume and quality of training data can be enhanced. In addition, we introduced a GAN-based data augmentation and balancing technique.

In this study, we focus on the design of the Offline Training Module. We propose a generic ML-based IDS that would be more robust against unseen data. We also demonstrate how a lack of data can impact the performance of the model when

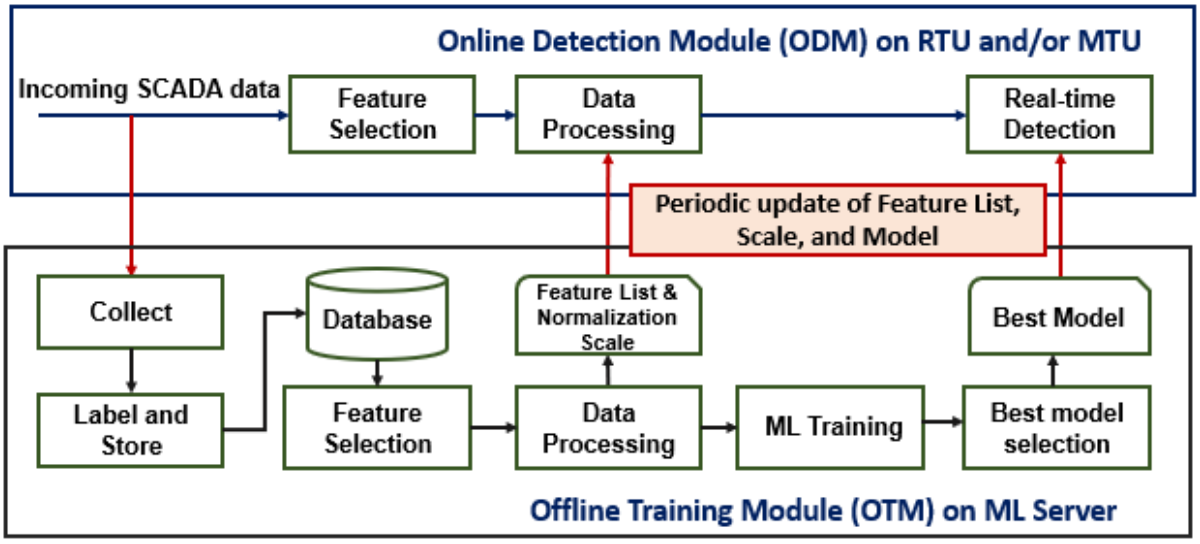

FIGURE 2. Process diagram of the proposed framework for intrusion detection in SCADA-based Power Systems.

tested with unseen data. Also, the need for balanced data for training in achieving better performance is highlighted in our research.

IV. DESCRIPTION OF DATASETS

In the past, numerous researchers have utilized the Oak Ridge National Laboratory (ORNL) dataset which was produced at Mississippi State University [16], that emulates a power transmission system testbed, thereby underscoring its significance. The summary table depicted in section II focuses on past research methods based on the ORNL dataset. To the best of our knowledge, the power system dataset available for such research is currently limited to the ORNL dataset. As a typical SCADA dataset for the power system is not publicly available, we, like other researchers [8], [9], [13], [14], [15], validated our framework using this dataset. However, our framework can easily accommodate other more realistic datasets for developing high-performance models.

The entire datasets are classified into three different classes, namely, binary-class, three-class, and multi-class. In this study, we are only focusing on binary classification, i.e., normal vs. intrusion. There is a total of fifteen files in ''.csv format'', i.e., comma-separated values format. Each of these ''.csv files'' contains 128 columns referred to as features and about 5000 records each. Out of these 128 features, 116 features are measurements related to electric signals and 12 features inserted by the PMU (Phasor Measurement Unit) are control panel logs, alerts from snort, and relays. Moreover, to obtain the 116 features, four phasor measurement units were used to measure the electric signals on the power grid. Each phasor measurement unit measures 29 features contributing to 116 features. The former 116 features are referred to as the Signal Reference usually represented by R followed by a number [R#] and it specifies their location in the PMU and type of measurements. Table 1 depicts the feature names and their corresponding description. The last column in every dataset is the output label (not shown in Table 1).

TABLE 2. Features of ORNL power system dataset.

Features	Description
PA1:VH-PA3:VH	Phase A-C Voltage Phase Angle
PM1:V-PM3:V	Phase A-C Voltage Magnitude
PA4:IH-PA6:IH	Phase A-C Current Phase Angle
PM4:I-PM6:I	Phase A-C Current Magnitude
PA7:VH-PA9:VH	Pos.-Neg.-Zero Voltage Phase Angle
PM7:V-PM12:V	Pos.-Neg.-Zero Voltage Magnitude
PA10:VH-PA12:VH	Pos.-Neg.-Zero Current Phase Angle
PM10:V-PM12:V	Pos.-Neg.-Zero Current Magnitude
F	Frequency for relays
DF	Frequency Delta (dF/dt) for relays
PA:Z	Apparent impedance seen by relays
PA:ZH	Apparent impedance Angle seen by relays
S	Status Flag for relays

V. SYNTHETIC DATA GENERATION AND FEATURE SELECTION

A. SYNTHETIC DATA GENERATION

GANs are a deep learning-based generative model that involves two neural networks (NNs), namely, generator and discriminator [27]. Synthetic data are generated using a generator neural network that uses random data to start training the network then passes through a discriminator that is trained with real and synthetic data to distinguish between the real data and synthetic data. Both NNs try to optimize a different and opposing objective function or loss function. This is essentially an actor-critic model. As the discriminator changes its behavior, so does the generator, and vice versa.

GANs have the ability to learn a complicated high-quality model to generate tabular data [28]. The Python libraries,

namely YData and SDV, provide a platform to generate and validate the quality of synthetic data. YData synthesizer includes several GAN architectures, namely, CGAN (Conditional GAN), WGAN (Wasserstein GAN), WGAN-GP (Wassertein GAN with Gradient Penalty), DRAGAN (On Convergence and stability of GANS), Cramer GAN (The Cramer Distance as a Solution to Biased Wasserstein Gradients), CWGAN-GP (Conditional Wassertein GAN with Gradient Penalty) [29]. SDV works on a recursive modeling concept that is more efficient in creating complex datasets using statistical properties and machine learning modeling. Furthermore, this tool provides the testing ability to validate the quality of synthetic datasets.

GAN models have shown to be quite effective in a wide range of machine learning applications, including tabular data generation. Some state-of-the-art models of tabular data generation include CT-GAN, TableGAN, and MedGAN which are based on GAN models. These models have resulted in superior performance in generating artificial data when trained on a range of datasets [6]. In this study, we used CT-GAN for generating synthetic data to balance the dataset with an equal number of normal and attack vectors. CT-GAN model has proven to be a good synthesizer for tabular datasets [29]. Here, we have briefly discussed the working of a Generative Adversarial Network (GAN) model for synthetic data generation in the context of tabular data. Generating data with handcrafted distributions is in wide use while synthesizing data using learned distributions is an area of a recent study [27]. This method can systematically address quantity (e.g., data imbalance issues), quality, and privacy issues by substituting real data with synthetic data.

B. FEATURE SELECTION

Feature selection plays a crucial role in enhancing the accuracy of estimators and improving model performance, particularly in high-dimensional datasets. One of the approaches for feature selection is the Recursive Feature Elimination (RFE) method, which recursively selects features by comparing the performance of a larger feature set with that of a smaller one during the training process [30]. RFE follows a greedy optimization approach, evaluating various feature combinations against performance metrics such as accuracy and false-positive rate. By assessing these metrics, RFE identifies the features that contribute to the best scores. The algorithm iteratively eliminates ''weak'' and irrelevant features, progressively refining the feature set [31]. Once the most promising features are identified, they can effectively be utilized for training and testing the model.

In our study, we employed a Random Forest classifier in conjunction with RFE. Random Forest is a versatile bagging ensemble learning algorithm suitable for both classification and regression tasks. It combines multiple decision trees to make predictions and provide feature importance measures. To evaluate the selection of recursive features using Random Forest, we have performed the following steps:

1) We train a Random Forest model on the complete set of features of the power system using the training data.
2) In the Random Forest model, we calculate the importance of each feature, which is also known as the Weighted Feature Importance (WFI) score. This importance score is determined by evaluating the average decrease in impurity (such as Gini impurity or entropy) that occurs when splitting on a particular feature across all trees in the forest.
3) We iteratively remove the least important features by comparing their scores and removing the 10 features with the lowest scores. The purpose of removing 10 features is to assess the performance of the model by progressively eliminating a consistent number of features that provide improved predictability.
4) Once we reach the desired number of features or the importance threshold has been met, the process is stopped. Otherwise, the algorithm returns to step 1 and repeats the process on the reduced feature set.

We implemented Recursive Feature Selection with Random Forest using the scikit-learn library of Python. In our implementation, the code creates a Random Forest classifier and initializes the RFE object with the Random Forest model as the estimator. The parameter *n_features_to_select* specifies the desired number of features to select. After fitting the RFE object to the data using the *fit_transform* method, the *get_support* method is utilized to retrieve the indices of the selected features. These indices can be used to subset the original feature set for further analysis or modeling purposes.

VI. MODEL DESIGN AND DEVELOPMENT

In this section, we have described our proposed framework which focuses on data processing and model assessment methodology. To improve the data processing, we have applied the concept of data augmentation. Further, for effective assessment, we have evaluated the model using two validation techniques, namely, cross-validation, and holdout validation. The development and performance evaluation of the proposed model is carried out using publicly available power system datasets. These datasets are generated at different locations of PMUs. However, in our proposed methodology, we have offered a generic framework by combining these datasets into one while training the model as well as by augmenting the dataset with synthetic data. Moreover, the assessment of the model is carried out using holdout validation on the best model found while training the model to ensure proper validation of the model. We set aside 20% of the augmented dataset for holdout validation.

The proposed methodology for data processing and analysis for IDS model development is depicted in Figure 3 (steps 1 to 7). At first, we combined the fifteen small datasets into one and extracted the 128 measurements along with binary output labels for our experiments as shown in step 1. It should be noted here that although the model performance on each individual dataset (training and testing from the same dataset) can be high, obtaining a highly performant model

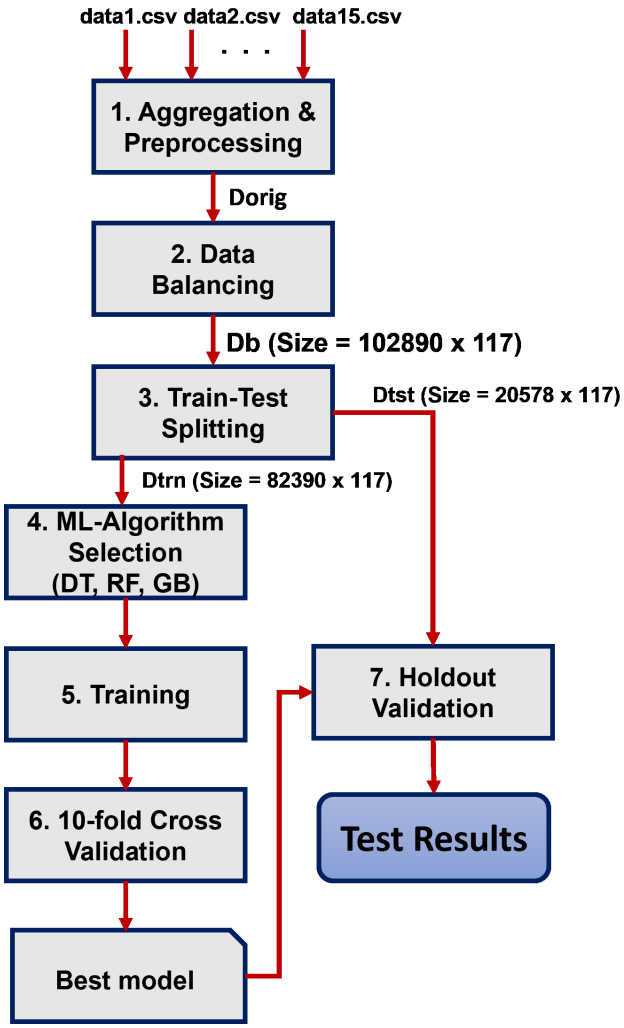

FIGURE 3. Data processing with augmentation and model evaluation methodology.

while training with the combined dataset and testing from any of these 15 datasets (not used in training) is not readily feasible as will be shown in next section. However, training a model with more data can definitely yield a more generic model.

Once the combined dataset with the binary label is generated, we removed the 12 log features out of 128 features from the combined dataset. This step was performed to make the model independent of other tools that collect logs including snort a traditional intrusion detection system. We also add one more column called datafile to indicate which file the data came from. However, this column information is not used in model training.

The balance dataset is one of the crucial factors while training the model. And, hence, we have applied data augmentation to balance the dataset as depicted in step 2. For that, we computed the class distribution, i.e., the normal to attack vector ratio. In this combined dataset, D_{comb}, we noticed about a 1:2 ratio between normal and attack vectors. Next, we generated synthetic normal data using

TABLE 3. Size of processed dataset.

Steps	Size	Notes
1	72073×118 (D_{comb})	One metadata column not used in model training
2	102890×119 (D_{aug})	Two metadata columns not used in model training
3	82390×119 (D_{trn}) 20578×119 (D_{tst})	Two metadata columns not used in model train-ing/testing

CT-GAN to balance the dataset in terms of normal and attack vectors. While data augmentation by generating synthetic data helped us generate a balanced dataset, it also improves the generalizability of the models by having a larger training dataset.

After data augmentation, we obtained 51445 normal and 51445 attack vectors with 116 measurement features and one label column. We also insert a new column indicating real (0) or synthetic (1) data. This new column in the dataset was only used in one of our experiments where we validated the efficacy of synthetic data generation using CT-GAN. We randomly split the augmented dataset, D_{aug} into two different datasets D_{trn} and D_{tst} for training and testing, respectively as shown in step 3. The testing data was set aside for testing only, while the training data was used for cross-validation. Models were trained using three tree-based machine learning algorithms, namely, Decision Tree (DT), Random Forest (RF), and Gradient Boosting (GB) as they are found to be quite accurate for these types of applications [14], [15]. The algorithm selection and training steps are shown in steps 4 and 5, respectively. The best model found during cross-validation was saved for testing with test data. The size of the processed dataset after each step is shown in Table 2.

To test the best model with the unseen data, we performed two sets of validation, namely, cross-validation and holdout validation as shown in steps 6 and 7. For our experiments, we also cross-validated models trained with the combined original dataset (i.e. before augmenting the dataset with synthetic data for class balancing) and one of the data files out of the fifteen total files in the ORNL power system dataset. Once the cross-validation was done, we also tested the best model found in the cross-validation stage with the test data D_{tst} that was set aside for the holdout validation. The trained models are saved during the cross-validation stage, i.e., implementing the off-line training module and tested using test data that was set aside to simulate the on-line testing module of the proposed design framework.

VII. EXPERIMENTS, RESULTS, AND DISCUSSIONS

The main component of the ML-based IDS is the classification models. These models need to be trained with high volume and high-quality data to obtain good performance which may not be always readily available. The framework designed here includes methodologies to augment (and also balance) real datasets and provide a means for effective evaluation. This section demonstrates how cross-validation can be misleading and how we test the model generated

TABLE 4. Validation of synthetic data generation using Logical Regression Classifier (using normal events).

Accuracy	F1-Score	Precision	Recall	FPR	FNR
0.562	0.561	0.562	0.562	0.438	0.438

TABLE 5. Cross-validation results with single dataset.

Alg.	Accuracy	F1-Score	Precision	Recall	FPR	FNR
DT	0.9433	0.9606	0.9588	0.9623	0.1049	0.0377
RF	0.9734	0.9816	0.9736	0.9897	0.0680	0.0103
GB	0.8852	0.9241	0.8799	0.9732	0.3377	0.0268

during cross-validation with unseen data to gain more confidence in the ML/DL IDS model. We have developed a tool in Python to implement these methodologies to realize the Offline Training Module and Online Detection Module of the IDS.

We describe the experiments conducted to validate our methodology of data processing, including data augmentation with synthetic data and model training and evaluation. Each experiment is described with the results and discussion.

A. EXPERIMENT 1: VALIDATION OF SYNTHETIC DATA GENERATION

The purpose of this experiment is to evaluate the quality of the synthetic data, more specifically to investigate how closely it matches with real data. For this assessment, we removed the original label column (i.e. normal vs. intrusion) in the augmented dataset and used the newly added column to indicate real vs. synthetic. Since we only used synthetic data for normal records, we removed all real intrusion data. The dataset now contains only normal real and synthetic data with the last column indicating the type. The results are tabulated in Table 4. We expected to see nearly 50% accuracy, as CT-GAN data attempts to generate data as similar as possible to the real data so it will be difficult to distinguish between real and synthetic data. Table 4 shows about 56.2% average accuracy while cross-validated using a linear-based classifier called Logical Regression, thus validating the efficacy of the CT-GAN model.

B. EXPERIMENT 2: CROSS-VALIDATION OF THE ML MODELS USING A SINGLE DATASET (FROM ONE OF THE FIFTEEN DATA FILES)

This experiment aims to select the best model based on cross-validation results of the ML models developed using a training dataset generated from one of the 15 files in the ORNL power system dataset. In this experiment, we used only one data file to develop three classification models using three different algorithms, namely, DT, RF, and GB. The 10-fold cross-validation results are shown in Table 5. We obtained RF as the best model during cross-validation. The best RF model found during cross-validation, M_{single_best} is saved for testing with new data with 20578 records.

TABLE 6. Cross-validation results with combined dataset.

Alg.	Accuracy	F1-Score	Precision	Recall	FPR	FNR
DT	0.8814	0.9167	0.9189	0.9145	0.2011	0.0854
RF	0.9358	0.9563	0.9297	0.9845	0.1857	0.0154
GB	0.7422	0.8448	0.7406	0.9831	0.8589	0.0169

TABLE 7. Cross-validation results with combined and augmented dataset.

Alg.	Accuracy	F1-Score	Precision	Recall	FPR	FNR
DT	0.8707	0.8710	0.8711	0.8709	0.1295	0.1291
RF	0.9395	0.9413	0.9152	0.9691	0.0902	0.0309
GB	0.8088	0.8353	0.7349	0.9676	0.3506	0.0324

C. EXPERIMENT 3: CROSS-VALIDATION OF THE ML MODELS USING A COMBINED FIFTEEN FILES INTO ONE DATASET

This experiment aims to select the best model based on cross-validation results of the ML models when trained with a training dataset generated from combining 15 files in ORNL power system dataset. In this experiment, we trained and developed three models with the combined dataset, namely, DT, RF, and GB. 10-fold cross-validation is used to evaluate model performance. Table 5 shows the results. The best RF model found during cross-validation, M_{comb_best} is saved for testing with new data with 20578 records.

D. EXPERIMENT 4: CROSS-VALIDATION OF THE ML MODELS USING COMBINED AND AUGMENTED TRAINING DATASET

The purpose of this experiment is to select the best model based on cross-validation results of the ML models when trained with a training dataset combining 15 files in ORNL dataset and augmenting it with synthetic data generated by CT-GAN. In this experiment, we trained and developed three models with augmented training datasets. 10-fold cross-validation was used to evaluate model performance. Table 6 shows the results. The best RF model found during cross-validation, M_{aug_best} is saved for testing with new data with 20578 records.

E. EXPERIMENT 5: TEST THE BEST MODEL FOUND IN EXPERIMENT 2 WITH UNSEEN DATA (SINGLE DATASET)

The purpose of this experiment is to investigate how the best model found in Experiment 2 (i.e., the model trained with a small dataset) performs when tested using unseen data. In this experiment, we tested the best model found in Experiment 2, i.e., M_{single_best} with the test dataset that was set aside. The result is shown in Figures 4 (a) and (b). From Figure 4, we can see that the performance of the M_{single_best} is not very satisfactory when tested with unseen data. Since the dataset, it trained with is smaller in size (4888 records), the model can not be generalized. The false positive is extremely high as shown in Figure 4 (b) (about 80%), making the model impractical for use although the cross-validation result was the best among all other models developed in this study.

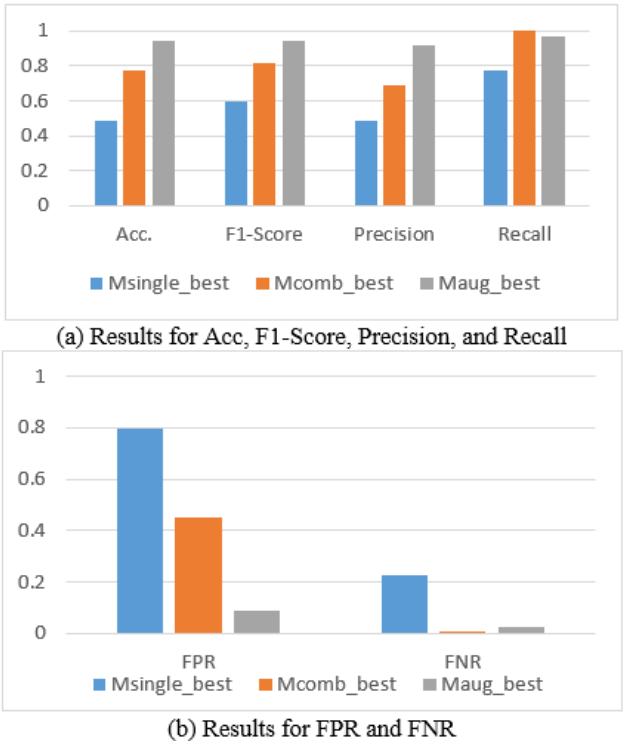

FIGURE 4. Model performance comparison with unseen data.

F. EXPERIMENT 6: TEST THE BEST MODEL FOUND IN EXPERIMENT 3 WITH UNSEEN DATA (COMBINED DATA SET)

The purpose of this experiment is to investigate how the best model found in Experiment 3 (i.e., the model trained with combined data) performs when tested using unseen data. In this experiment, we tested the best model found in Experiment 4, i.e., M_{comb_best} with the test dataset that was set aside. The results are shown in Figures 4 (a) and (b). The model did not perform well especially since it exhibited a very high false positive rate of about 44%. However, the detection rate or recall was very high as also indicated by only 0.12% false negative rate or miss rate.

G. EXPERIMENT 7: TEST THE BEST MODEL FOUND IN EXPERIMENT 4 WITH UNSEEN DATA (COMBINED AND AUGMENTED DATASET)

The purpose of this experiment is to investigate how the best model found in Experiment 4, i.e., the model trained with the combined and augmented dataset) performs when tested using unseen data. In this experiment, we tested the best model found in Experiment 6, i.e., M_{aug_best} with the test dataset that was set aside. The result is shown in Figures 4 (a) and (b). Overall this model performed the best if we compare the F1-Score value as shown in Figure 4 (a). However, this model did not perform well in terms of recall as also observed by its higher false negative rate than that of M_{comb_best} as shown in Figure 4 (b).

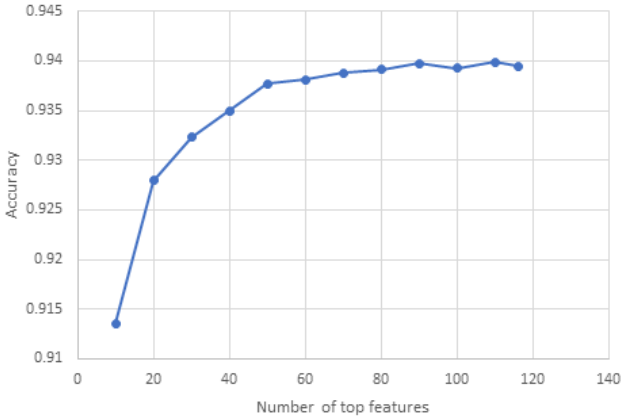

FIGURE 5. Accuracy with feature selection.

TABLE 8. Results of feature selection (RFE-RF) of combined augmented dataset.

Features	Accuracy	F1-Score	Recall	FPR	FNR
10	0.9136	0.9172	0.9551	0.1282	0.0449
20	0.9279	0.9305	0.9631	0.1074	0.0369
30	0.9323	0.9346	0.9652	0.1008	0.0348
40	0.9350	0.9372	0.9678	0.0980	0.0322
50	0.9377	0.9397	0.9693	0.0941	0.0307
60	0.9381	0.9401	0.9691	0.0930	0.0309
70	0.9388	0.9407	0.9684	0.0910	0.0316
80	0.9391	0.9410	0.9685	0.0903	0.0315
90	0.9397	0.9416	0.9691	0.0898	0.0309
100	0.9393	0.9412	0.9698	0.0914	0.0302
110	0.9399	0.9418	0.9703	0.0906	0.0297
116	0.9395	0.9413	0.9691	0.0902	0.0309

It is worth mentioning that the Random Forest-based models even with default hyperparameters perform the best in all our experiments. Next, we performed automated feature selection to make the model more computationally efficient and more privacy-preserving.

H. EXPERIMENT 8: CROSS-VALIDATION OF MODELS USING COMBINED AND AUGMENTED TRAIN DATASET AND WITH FEATURE SELECTION

This experiment aims to find out whether a reduced feature set can provide a better model than when using all features. During the assessment, we applied Recursive Feature Elimination with Random Forest (RFE-RF) approach to select the top N features and created the model using only the selected features. N was varied from 10 to 110 and cross-validation of each model was performed with an augmented train dataset. The results from this experiment are shown in Table 8. From Table 8, it was observed that the performance of the model improves as more features are included, however, the rate of change in accuracy and other performance metrics is not significant after N=50 as shown in Figure 5. Also, although the best accuracy of 93.99% is found with the top 110 features included, N=90 provides the lowest false positive rate of 8.98%. It should be noted that although feature selection does

TABLE 9. Best model performance with feature selection (RFE-RF) on unseen data (TOP 50 features-39% of total features).

Accuracy	F1-Score	Precision	Recall	FPR	FNR
0.9397	0.9409	0.9152	0.9683	0.0884	0.0317

not always result in better performance, this step is crucial and desirable when a privacy-preserving model is required and/or training and inference time are of great concern. Building models with fewer features can also help in reducing the complexity and enhance the interpretability of the model. This will also minimize data storage and communication costs.

I. EXPERIMENT 9: TEST THE BEST MODEL FOUND IN EXPERIMENT 7 WITH UNSEEN TEST DATA USING 50 MOST USEFUL FEATURES

This experiment aims to evaluate further the best model found in Experiment 8 with unseen data. The results of this experiment for models with the top 50 features are shown in Table 9. As seen from Table 9, the model with the top 50 features performs quite satisfactorily with unseen data with an accuracy of 93.97% and F1-Score of 94.09%, slightly better than the cross-validation results.

Finally, we compared our best models with and without feature selection with some published models that were trained with the ORNL power system dataset. Table 10 shows the comparison results for accuracy. It is worth noting that the previous models were not trained with the combined dataset using all 15 data files and were not tested with unseen data. Although our model performance is not the best when compared with the cross-validation results of other models, it is more generalized and therefore works reasonably well when tested with unseen data.

J. KEY TAKEAWAYS AND FUTURE DIRECTIONS

Although machine learning methodologies and modeling are widely exercised in the research community, they are not yet commonly implemented in practice. In intrusion detection applications, one of the main barriers is the high false positive rates. One of the challenges in achieving a high-performing model is the lack of high-volume and high-quality data. Data imbalance is another issue, as normal incidents typically outnumber intrusion records. However, in the case of data collected in a controlled environment, such as in the ORNL dataset, we observed a roughly 1:2 ratio of normal to intrusion records. Synthetic data generation techniques using GANs are gaining popularity for augmenting real data to increase the training dataset's volume and balance. We achieved significantly improved performance when training with an augmented dataset and testing the model on unseen data. We demonstrated that cross-validation results, especially from models trained with small datasets, can often be misleading and may not generalize well to unseen data. In our case, the model trained with one of the fifteen ORNL

TABLE 10. Performance comparison of IDS models trained with ORNL power system data (Combined - Using all 15 files for training or not).

Classifier	Feature %	Combined Dataset	Accuracy
ADA-JRIP [8]	100%	No	94.55%
EMCT [11]	25%	No	70.60%
GMMKM [13]	25%	No	94.56%
Tree Based [14]	12%	No	97.66%
Majority Vote [15]	23%	No	98.24%
RF (proposed)	**100%**	**Yes**	**93.95%**
RFE-RF (proposed)	**39%**	**Yes**	**93.97%**

datasets showed the best cross-validation performance but the worst performance when tested with unseen data. In our ML/DL model design framework, we fine-tune the model by incorporating more synthetic records into the original dataset and validate the model in two different ways to gain greater confidence in the final model. This approach closely mimics the real deployment pipeline, involving both offline training and online inference.

We still believe there is ample scope for improvement in the methodology proposed for the ML/DL model design framework. For example, we cannot guarantee the CT-GAN-generated data will meet the constraints that the electrical measurements should maintain. For example, the data must be in accordance with circuit theory, for instance, the algebraic sum of all currents in a busbar should be zero (Kirchhoff's current law). However, the quality of the synthetic data was measured using two techniques, namely, statistical analysis and Machine Learning modeling. Further details about these techniques can be found in [32]. Moreover, an improvement in the validation methodology will include removing synthetic data from the test dataset. The work presented in this paper lays a solid foundation for working towards a more improved ML/DL model-based IDS design framework for the power grid. As part of future work, we are also looking into collecting data from the testbed that simulates closely the real power system. The work is currently underway with an aim to simulate up to 2000 three-phase buses with the co-simulation setup supported by RTDS - the world standard for real-time digital power system simulation.

VIII. CONCLUSION

In this paper, we proposed and evaluated a generic framework for designing a machine learning-based Intrusion Detection System (IDS) for SCADA-based power systems. In the offline training module of our proposed framework, we combined datasets from different sources, specifically 15 data files. We then augmented and balanced the training dataset by generating synthetic data using CT-GAN. We evaluated and compared three classical machine learning algorithms (Decision Tree, Random Forest, and Gradient Boosting) using both cross-validation and holdout validation techniques to identify a high-performing and generic intrusion detection model for a sample power grid SCADA system. The results demonstrated that the models created with augmented

data outperformed while training with the real dataset. By combining real and synthetic data for training, and by utilizing only selected useful features, we were able to create a more generic model. We achieved promising results by applying feature selection (RFE-RF) and testing the model with unseen data on the 50 most significant features. We obtained an accuracy of 93.97% when tested with an unseen dataset.

REFERENCES

[1] D. Upadhyay and S. Sampalli, "SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability assessment and security recommendations," *Comput. Secur.*, vol. 89, Feb. 2020, Art. no. 101666.

[2] A. Peterson. (2016). *Should You be Afraid of Cyberattacks on Nuclear Power Plants?*. Washington Post. [Online]. Available: https://www.washingtonpost.com/news/the-switch/wp/2016/01/15/should-you-be-afraid-of-cyberattacks-on-nuclear-power-plants/

[3] SANS, Elections Infrastructure Information Sharing & Analysis Center. *Analysis of the Cyber Attack on the Ukrainian Power Grid*. Accessed: Jun. 8, 2023. [Online]. Available: https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2016/05/20081514/E-ISAC_SANS_Ukraine_DUC_5.pdf

[4] J. Tidy. (2021). *Colonial Hack: How did Cyber-Attackers Shut Off Pipeline?*. BBC News. [Online]. Available: https://www.bbc.com/news/technology-57063636

[5] N. Kshetri and J. Voas, "Hacking power grids: A current problem," *Computer*, vol. 50, no. 12, pp. 91–95, Dec. 2017.

[6] Y. Zhang, N. A. Zaidi, J. Zhou, and G. Li, "GANBLR: A tabular data generation model," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Dec. 2021, pp. 181–190.

[7] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–11.

[8] R. C. B. Hink, J. M. Beaver, M. A. Buckner, T. Morris, U. Adhikari, and S. Pan, "Machine learning for power system disturbance and cyber-attack discrimination," in *Proc. 7th Int. Symp. Resilient Control Syst. (ISRCS)*, Aug. 2014, pp. 1–8.

[9] S. Pan, T. Morris, and U. Adhikari, "Classification of disturbances and cyber-attacks in power systems using heterogeneous time-synchronized data," *IEEE Trans. Ind. Informat.*, vol. 11, no. 3, pp. 650–662, Jun. 2015.

[10] S. Pan, T. Morris, and U. Adhikari, "Developing a hybrid intrusion detection system using data mining for power systems," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 3104–3113, Nov. 2015.

[11] M. Keshk, N. Moustafa, E. Sitnikova, and G. Creech, "Privacy preservation intrusion detection technique for SCADA systems," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2017, pp. 1–6.

[12] N. Moustafa, E. Adi, B. Turnbull, and J. Hu, "A new threat intelligence scheme for safeguarding Industry 4.0 systems," *IEEE Access*, vol. 6, pp. 32910–32924, 2018.

[13] M. Keshk, E. Sitnikova, N. Moustafa, J. Hu, and I. Khalil, "An integrated framework for privacy-preserving based anomaly detection for cyber-physical systems," *IEEE Trans. Sustain. Comput.*, vol. 6, no. 1, pp. 66–79, Jan. 2021.

[14] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, "Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 1, pp. 1104–1116, Mar. 2021.

[15] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, "Intrusion detection in SCADA based power grids: Recursive feature elimination model with majority vote ensemble algorithm," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2559–2574, Jul. 2021.

[16] U. Adhikari, S. Pan, T. Morris, R. Borges, and J. Beaver. Industrial Control System (ICS) Cyber Attack Datasets. Datasets Used in the Experimentation. Accessed: May 4, 2023. [Online]. Available: https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets

[17] M. Krishnan, S. Gugercin, and P. A. Tarazaga, "A wavelet-based dynamic mode decomposition for modeling mechanical systems from partial observations," *Mech. Syst. Signal Process.*, vol. 187, Mar. 2023, Art. no. 109919.

[18] V. Saravanan, M. Ramachandran, and M. Selvam, "Interaction between technical and economic benefits in distributed generation," *Electr. Autom. Eng.*, vol. 1, no. 2, pp. 83–91, Jul. 2022.

[19] M. Umer, S. Sadiq, H. Karamti, R. M. Alhebshi, K. Alnowaiser, A. A. Eshmawi, H. Song, and I. Ashraf, "Deep learning-based intrusion detection methods in cyber-physical systems: Challenges and future trends," *Electronics*, vol. 11, no. 20, p. 3326, Oct. 2022.

[20] J. Wang, J. Pan, I. AlQerm, and Y. Liu, "Def-IDS: An ensemble defense mechanism against adversarial attacks for deep learning-based network intrusion detection," in *Proc. Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2021, pp. 1–9.

[21] M. H. L. Louk and B. A. Tama, "Revisiting gradient boosting-based approaches for learning imbalanced data: A case of anomaly detection on power grids," *Big Data Cogn. Comput.*, vol. 6, no. 2, p. 41, Apr. 2022.

[22] H. Karimipour, S. Geris, A. Dehghantanha, and H. Leung, "Intelligent anomaly detection for large-scale smart grids," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2019, pp. 1–4.

[23] I. Dragos. *2020 ICS Cybersecurity Year in Review*. Accessed: Jun. 4, 2023. [Online]. Available: https://www.dragos.com/blog/industry-news/2020-ics-cybersecurity-year-in-review/

[24] L. Moniz, A. L. Buczak, B. Baugher, E. Guven, and J.-P. Chretien, "Predicting influenza with dynamical methods," *BMC Med. Informat. Decis. Making*, vol. 16, no. 1, pp. 1–17, Dec. 2016.

[25] S. D. Roy, S. Debbarma, and A. Iqbal, "A decentralized intrusion detection system for security of generation control," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 18924–18933, Oct. 2022.

[26] T. Yang, Y. Liu, and W. Li, "Attack and defence methods in cyber-physical power system," *IET Energy Syst. Integr.*, vol. 4, no. 2, pp. 159–170, Jun. 2022.

[27] L. Xu and K. Veeramachaneni, "Synthesizing tabular data using generative adversarial networks," 2018, *arXiv:1811.11264*.

[28] M. H. Shahriar, N. I. Haque, M. A. Rahman, and M. Alonso, "G-IDS: Generative adversarial networks assisted intrusion detection system," in *Proc. IEEE 44th Annu. Comput., Softw., Appl. Conf. (COMPSAC)*. Los Alamitos, CA, USA: IEEE, Jul. 2020, pp. 376–385, doi: 10.1109/COMPSAC48688.2020.0-218.

[29] M. License. *YData Synthetic*. Accessed: Jul. 5, 2023. [Online]. Available: https://github.com/ydataai/ydata-synthetic

[30] B. F. Darst, K. C. Malecki, and C. D. Engelman, "Using recursive feature elimination in random forest to account for correlated variables in high dimensional data," *BMC Genet.*, vol. 19, no. S1, pp. 1–6, Sep. 2018.

[31] C. Zhang, Y. Li, Z. Yu, and F. Tian, "Feature selection of power system transient stability assessment based on random forest and recursive feature elimination," in *Proc. IEEE PES Asia–Pacific Power Energy Eng. Conf. (APPEEC)*, Oct. 2016, pp. 1264–1268.

[32] D. Upadhyay, Q. Luo, J. Manero, M. Zaman, and S. Sampalli, "Comparative analysis of tabular generative adversarial network (GAN) models for generation and validation of power grid synthetic datasets," in *Proc. 20th Int. Conf. Smart Technol. (EUROCON)*, Jul. 2023, pp. 677–682.

MARZIA ZAMAN received the M.Sc. and Ph.D. degrees in electrical and computer engineering from the Memorial University of Newfoundland, Canada, in 1993 and 1996, respectively. She started her career at Nortel Networks, Ottawa, Canada, in 1996, where she joined the Software Engineering Analysis Laboratory (SEAL), and later joined the Optera Packet Core Project, as a Software Developer. In addition, she has many years of industry experience, as a Researcher and a Software Designer with Accelight Networks, Excelocity, Sanstream Technology, and Cistel Technology Inc. Since 2009, she has been with the Centre for Energy and Power Electronics Research, Queen's University, Canada, and one of its industry collaborators, Cistel Technology Inc., on multiple power engineering projects. Her research interests include renewable energy, wireless communication, the IoT, cyber security, machine learning, and software engineering.

DARSHANA UPADHYAY received the master's degree in computer science from Nirma University, India, and the Ph.D. degree from the Faculty of Computer Science, Dalhousie University, Canada. She was an Assistant Professor with Nirma University. She is currently a Postdoctoral Fellow with Cistel Technology Inc., under the supervision of Dr. Sampalli and Dr. Marzia Zaman. Her research interests include network and information security, algorithm conceptualization, hardware design in the field of embedded systems, vulnerability assessments, and intrusion detection and prevention techniques for the IoT/SCADA-based systems. During her graduate studies, she was awarded the Gold Medal for securing the First position. She was a co-recipient of the Indo-Canadian Shastri Research Grant in the field of secure mobile communication. In addition, she was selected as one of the Canada's 2020 Emerging Thought Leaders by the Women in International Security-CANADA. She received the 2021 Citizenship Award and the 2022 Leadership Award from the Faculty of Computer Science, Dalhousie University.

CHUNG-HORNG LUNG (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer science and engineering from Arizona State University, in 1988 and 1994, respectively. He was with Nortel Networks, from 1995 to 2001. At Nortel Networks, he was with the Software Engineering Analysis Laboratory (SEAL), focusing on software architecture/performance engineering and re-engineering, and optical packet interworking (OPi), and on network traffic engineering with MPLS protocol. In September 2001, he joined the Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada, where he is currently a Full Professor. He received various research grants from NSERC, OCE, MITACS, CFI, and NCIT. He was closely with industrial partners, including Ciena, Cisco, Cistel Technology Inc., Ericsson, Huawei, and Nokia. His research interests include computer networks, wireless communications, Ad Hoc and sensor networks, software/system engineering, and data analytics. He served as an organizing committee member or a technical program committee member for a number of international conferences, including IEEE ICC, IEEE GLOBECOM, and IEEE COMPSAC. He received several prestigious awards at Nortel, including the President's Award in the Quality Category (Team), in 1996, the Vice President's Award of Excellence, Advanced Software and Network Technology, in 1998, and the President's Award for Leadership in Technology (Team), in 1999.