Intrusion Detection in SCADA Based Power Grids: Recursive Feature Elimination Model With Majority Vote Ensemble Algorithm

Darshana Upadhyay, Jaume Manero, Marzia Zaman, and Srinivas Sampalli, *Member, IEEE*

***Abstract*—We propose an integrated framework for an intrusion detection system for SCADA (Supervisory Control and Data Acquisition)-based power grids. Our scheme combines RFE-XGBoost (Recursive Feature Elimination-eXtreme Gradient Boosting) based feature selection with a majority vote ensemble method. RFE selects features recursively based on Weighted Feature Importance (WFI) scores during the training process, while the majority vote ensemble method predicts the output label based on a total of nine heterogeneous classifiers - three bagging ensembles, namely, Random Forest (RF), Extra Tree (ET), and Decision Tree (DT), three boosting ensembles, namely, XGBoost (XGB), Gradient Boosting (GB), and AdaBoost-Decision Tree (AdB-DT) along with artificial neural network (ANN), Naive Bayes (NB), and k-nearest neighbors (KNN). This leads to a more accurate solution as a result of the combination of the most useful features and prediction from multiple heterogeneous classifiers. Experimental results show that our approach increases the accuracy, precision, recall, F1 score, and decreases the miss rate as compared to previous approaches. The model is also evaluated for four different class categories, namely binary, three-class, seven class and multi-class, using Precision Recall (PR) and Receiver Operating Characteristic (ROC) plot. In addition, an end-to-end IDS framework is proposed for efficient and accurate detection of intrusions.**

***Index Terms*—SCADA systems, power grids, recursive feature elimination, majority vote, ensemble method, feature selection, cyber security, network intrusions.**

I. Introduction

POWER grids are the underlying infrastructure that support our economies and daily lives by providing and sustaining a continuous supply of electricity. They play a fundamental role in connecting industries and homes with far away locations from where the power is originally generated. Furthermore, they assure the quality of the electricity supply at the point of consumption. In the past, power grids were isolated systems. The field devices of such systems were managed locally on the plant floor. However, as technology advanced, energy system devices were gradually monitored and controlled remotely. Currently, SCADA (Supervisory Control and Data Acquisition) systems play a vital role in the management of power grid components efficiently.

Current power grids comprise of multiple substations and control centers and widely spread in large geographical areas. Each substation consists of various components such as power lines, transformers, sensors, actuators, and phasor measurement units (PMUs), along with supervisory control and data acquisition (SCADA) elements for monitoring the system components remotely. Fig. 1 shows the block diagram of a SCADA architecture for power systems. A SCADA network segment typically includes a SCADA master, HMI (Human Machine Interface), and data historian placed at the control center, communication links, and various field control devices such as Programmable Logic Controllers (PLCs), Remote Terminal Units (RTUs), and IEDs (Intelligent Eletronic Devices). The sensors and actuators located at power grids frequently supply digital status information to the field control devices. These devices further communicate this information to MTU, where the server will process the data according to acceptable parameter ranges. This information will then be transmitted back to field control device to improve the performance and to avoid hazards. The SCADA master also stores the status information on the data historian and displays it on the HMI for centralized control and monitoring of the power grids.

However, this evolution has connected power systems to the Internet, which, in turn, can expose them to various cyber-attacks such as False Data Injection (FDI) attacks, Denial of Service (DoS), or Man-In-the-Middle (MIM) attacks [1], [2]. FDI manipulates the energy measurement parameters, either by identifying the backdoors that bypass the system or by using privileges of authorized personnel [3]. The cyber attack against the Ukrainian power plant in 2015 is one example of such attacks in which nearly 250,000 people were left without electricity for many hours [4]. Another example is the attack on the Davis-Besse nuclear power plant in Oak Harbor, USA [5] which was infected by the SQL Slammer worm. The worm infected the entire power system with a DoS attack launched

Manuscript received January 26, 2021; revised May 29, 2021; accepted July 15, 2021. Date of publication July 26, 2021; date of current version September 16, 2021. This work was supported by the Natural Sciences and Engineering Research Council (NSERC), Canada through a Collaborative Research Grant. Recommended for acceptance by Dr. Fan Wu. *(Corresponding author: Srinivas Sampalli.)*

Darshana Upadhyay and Srinivas Sampalli are with the Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 4R2, Canada (e-mail: darshana@dal.ca; srini@cs.dal.ca).

Jaume Manero is with the Technical University of Catalonia, 08034 Barcelona, Spain, and also with the Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 4R2, Canada (e-mail: jaume.manero@dal.ca).

Marzia Zaman is with the Research & Development Department, Cistel Technology Inc., Ottawa, ON K2E 7V7, Canada (e-mail: marzia@cistel.com).

Digital Object Identifier 10.1109/TNSE.2021.3099371

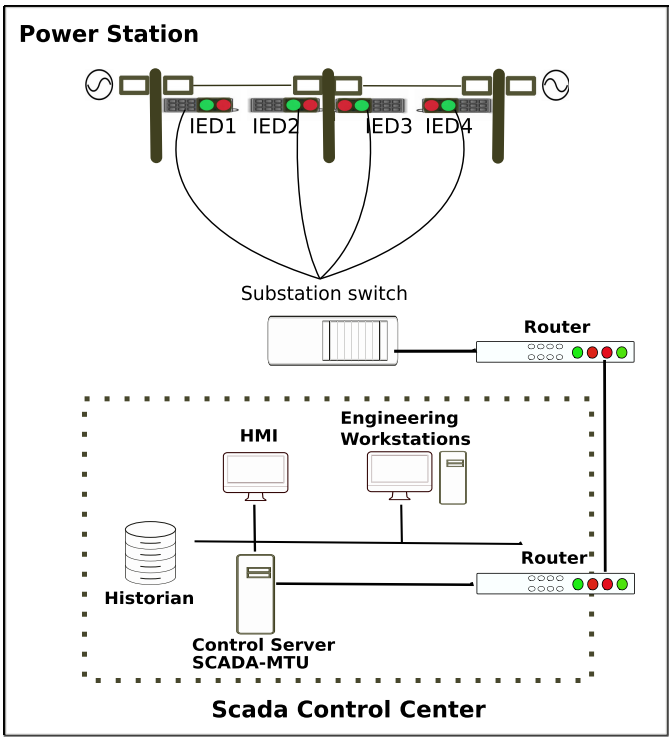


Fig. 1. SCADA System Architecture for Power System. Legend: HMI: Human Machine Interface, IEDs: Intelligent Electronic Devices, MTU: Master Terminal Unit.

by exploiting the vulnerabilities of the SCADA system. Such attacks on a nation's power grid can lead to catastrophic consequences [6].

Power grids face significant challenges pertaining to the security and privacy of the data. One of the challenges in securing power systems is the deployment of safeguards and the management of the network because of legacy-inherited security weaknesses and limitations. Although many security controls including defense in depth architecture, access control, authentication mechanisms, confidentiality, integrity techniques, and firewalls have been developed to protect critical infrastructure, the rapid evolution of hacking techniques can easily expose the integrity of the system's data and devices [7]. For example, in March 2019, the operators at a power grid center in the US lost communication with multiple sites of power generators due to a known firewall vulnerability [8].

Researchers have proposed intrusion detection techniques to secure SCADA based power grids. Hink *et al.* provide a comparative analysis of various machine learning techniques using a power grids dataset and identify Adaboost-JRIP as one of the best classifiers [9]. However, the authors do not filter and reduce the dimension of the dataset. Hence, they are unable to achieve good accuracy and execution speed. Pan *et al.* have focused on hybrid IDS using data mining, where they have used common path mining to identify the location of attacks [10], [11]. Further, in [12], the authors apply Pearson Correlation Coefficient (PCC) for feature selection and extract 75% of features. They use an Expectation Maximization Clustering Technique (EMCT) to classify the events. Using this approach, they improve the execution speed but do not achieve better accuracy for a multi-class dataset. Moreover, this technique is enhanced by combining PCC with the Gaussian Mixture - Kalman Filter Model (GMM-KF) in [14]. The authors are able to reduce the percentage of the features to 25 and achieve good accuracy and execution speed. However, this experiment is limited to a binary dataset. Moustafa *et. al.* [13] have used ICA - Independent Component Analysis feature selection and Beta Mixture Hidden Markov (BMHM) classification model. The authors have obtained promising results in regards to accuracy. However, they have worked on a subset of the features, and hence we could not identify the exact number of features used in this paper. We have recently proposed WFI based GBFS model for feature selection and extracted 12% of the most promising features in [15]. Our target was to achieve high execution speed and a better predictive model for real-time SCADA communication. The proposed GBFS model has been further verified with different machine learning algorithms. We have identified that the proposed solution is suitable for tree-based classifiers. Note that all these experimental studies use the power grid dataset created by Oak Ridge National Laboratories (ORNL). Table I summarizes the literature on IDSs for power grids.

In our earlier work [15], we have proposed a computationally efficient intrusion detection framework for power grids, which not only improves the computational cost but also provides privacy preservation. In that approach, we have determined the most significant features using a Weighted Feature Importance (WFI) based gradient boosting scoring model [15]. Furthermore, we have applied eight tree-based algorithms on multilevel multiple datasets to classify various attacks and normal events to validate the efficiency of derived features [15]. In particular, the most promising features were detected by considering multiple values of number of trees while training the model to apply the WFI scoring concept. From our preliminary results, we have identified three bagging ensembles, namely, Random Forest (RF), Extra Tree (ET), and Decision Tree (DT), three boosting ensembles XGBoost (XGB), Gradient Boosting (GB) and Ada-Boost-Decision Tree (AdB-DT) as the most promising classifiers. Moreover, we have identified the accuracy of other machine learning classifiers such as artificial neural network (ANN), Naive Bayes (NB), and k-nearest neighbors (KNN).

In this paper, we have enhanced the feature selection and classification module. The feature selection approach is extended by incorporating Recursive Feature Elimination (RFE) method. In that, the GBFS model is improved by replacing the gradient boosting with XGBoost as we found XGBoost is the most promising classifier amongst all the tree-based classifiers in our previous work. Hence, XGBoost can be a better fit to score the features using the WFI technique while training the dataset. Moreover, we have replaced the concept of evaluating number of trees while training the model with RFE approach. This approach helps us achieve a better predictive model by searching all the stable features instead of the most promising features while constructing the tree.

Another enhancement has been applied to the classification model by using a majority vote based ensemble method consisting of six tree-based classifiers along with artificial neural network (ANN), Naive Bayes (NB), and k-nearest neighbors

TABLE I
LITERATURE REVIEW OF PUBLISHED INTRUSION DETECTION SYSTEMS FOR POWER GRIDS

Attributes	Machine Learning for Power System Disturbance and Cyber-attack Discrimination [9]	Developing a Hybrid Intrusion Detection System Using Data Mining for Power Systems [10], [11]	Machine Learning for Power System Disturbance and Cyber-attack Discrimination [12]	A New Threat Intelligence Scheme for Safeguarding Industry 4.0 Systems [13]	An integrated framework for privacy-preserving based anomaly detection for cyber-physical systems [14]	Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids [15]
Feature Section Method	Not Applied	Not Applied	PCC (Pearson's correlation coefficient)	ICA (Independent component analysis)	PCC	GBFS
Features (%)	100%	100%	75%	compared various subset of features	25%	12%
Classification Technique	Adaboost - JRIP	Common Path Mining	Expectation Maximization Clustering Algo	Beta mixture-hidden Markov models (MHMMs)	Gaussian Mixture-Kalman Filter Model (GMM-KF)	Tree-Based (XGBoost)
Algorithm used	Machine Learning	Data mining and pattern recognition	Maximum likelihood estimation	Statistical common estimation method	Bayesian filtering algorithm	Machine Learning
Measure used	Accuracy	Accuracy	Accuracy, Precision, Recall, F-measure	Accuracy, Precision, Recall, F-measure	Accuracy, Precision, Recall, F-measure	Accuracy, Precision, Recall, F-measure, Training time
Dataset used	Oak Ridge National Laboratories (ORNL) - power grid dataset [16]					
Pros & Cons	Data are not pre-processed properly, Low accuracy and execution speed	Improvement in identification of attacks, provide location of attack.Moderate accuracy and execution speed	Improved execution speed but compromising in accuracy with multi-class datasets	Improved accuracy, no observations regarding execution speed	Improvement in accuracy, Tested for binary classification, No multi attack vectors classification	Significantly improved accuracy and execution speed

(KNN). We have replaced the single tree base classifier with a majority vote based ensemble method. The selection of various heterogeneous classifiers is based on our preliminary results [15]. This approach will determine the output label based on the majority of the class labels predicted by all the nine classifiers. The selected nine algorithms in this model work on different analogies, such as tree based, naive based, lazy learner, and neural networks based prediction. Consequently, the output label is calculated using a majority of heterogeneous predictions, which turned into a robust predictive model.

The concept of voting is used to average the output values based on the prediction of different classifiers. This process produces relatively uncorrelated output predictions of various classifiers which significantly reduces the error rate. Moreover, if output labels are highly correlated, in that case also this approach can easily detect a minor error. Furthermore, decision tree-based classifiers are good candidates for this approach as small perturbations generate totally different structures and splits. Hence combining prediction of such models using majority vote significantly improves the efficiency of the classification process. However, the execution speed and training time of this model could be higher than the single classifier. Hence, we have suggested an end to end machine learning based Intrusion Detection System framework for power grid SCADA security which utilize both the models as depicted in Figure 9 and described in Section VII. The objective and major contributions of this work are listed below.

Objective: The aim of this work is to propose a robust intrusion detection system for power grids which is compatible with time critical systems and has the capability to detect intrusions accurately using effective features of the network traffic. Moreover, the proposed model is a good fit for the control center to serve large-scale SCADA systems.

Contributions:

1) We use RFE-XGBoost based weighted feature importance scoring model to identify the most promising features. RFE selects the features recursively based on the weighted importance score of each feature by comparing a previously trained model with the current model. Through this approach, the most stable features of the dataset are determined which will be useful to achieve a better predictive model.
2) We derive 30 most promising features out of 128 features of the binary class, which significantly reduces the dimension of the dataset. Furthermore, the same features are used to the rest of the three categories, namely, three class, seven class, and multi-class to train the model to evaluate the efficiency of the RFE based feature selection model.
3) For the performance improvement, we apply the majority vote ensemble algorithm by considering nine heterogeneous classifiers to predict the output based on the majority of the class labels predicted by each of these nine classifiers.
4) We propose a deployment model of the IDS in SCADA-based power grids which reflects real-time traffic

monitoring by introducing placement of IDS models at the different locations, namely, plant floor, and control center.

For performance assessment and validation, we compare the accuracy of a total of nine classifiers along with the majority vote ensemble classifier. Moreover, we examine one of the classifiers of each method of bagging, boosting, and voting ensembles in terms of Precision-Recall (PR) and Receiver Operating Characteristic (ROC) plot. To validate the efficiency of the selected features and majority vote classifier, we evaluate the various performance metrics, namely, precision, recall, F1 score and miss rate of our proposed scheme. We also compare the accuracy of the majority vote ensemble method with existing bagging and boosting based ensemble techniques. Further, we compare the accuracy of the proposed methodology with published state-of-the-art techniques.

The rest of the paper is structured as follows. Section II describes the background and related work in the area of power grid security. The proposed intrusion detection framework and process diagram are described in Section III. Section IV covers algorithm conceptualisation and mathematical proof of RFE based feature selection and the majority vote ensemble method. Section V describes the experimental results and discussions. The proposed placement of IDS framework in SCADA based power grids is described in Section VI. Concluding remarks are provided in Section VII.

II. Background and Related Work

A. Power Grid Intrusion Detection Systems

The development of power grids has motivated researchers to propose various types of intrusion detection techniques to ensure security [17]. Generally, an IDS can be classified into two categories, namely, host-based and network-based. Host-based IDSs monitor the hosts on a network by collecting and monitoring various event logs of targeted devices. For example, a host based IDS for SCADA systems focuses explicitly on securing components such as RTUs and IEDs [12]. The IDS is responsible for identifying attacks against an IED of the substation by recording sequential events [18]. Network-based IDSs monitor the entire network traffic to detect malicious activities. This type of IDS can be further categorized into rule-based and anomaly-based IDS [19]. Rule-based IDSs are used in SCADA power grids for in-depth protocol analysis. This model works on the signature-based approach for pattern matching to analyze the input data for malicious packets [20]. In this approach, the signature of every incoming packet is compared with all the stored signatures to detect the threats. However, this approach works mainly for known threats but is unable to detect zero-day attacks [20]. Furthermore, the anomaly is detected based on packet loggers and packet sniffing tools to match the incoming traffic. This method is also less efficient for unidentified traffic [21].

More recently, data mining, clustering, data visualization, and statistical signal processing approaches have been used for intrusion detection. These techniques are more effective than rule-based intrusion detection, but typically produce a high level of false alerts [22]. Therefore, there is a need for more sophisticated methods that deal with real-time traffic monitoring. Machine learning-based techniques such as K-nearest neighbor (KNN), Hidden Markov models, and Support Vector Machines (SVM) have been used for detecting intrusion from real incoming traffic. KNN, also known as lazy learner, learns from nearest neighbors at run time [23]. However, this approach may be overfit for imbalanced small datasets. The support vector machine maps the input into another dimensional space, which offers promising results but is costly to train. Both these techniques require learning of expected anomaly but are sensitive to noise presented in the training datasets [24]. Similarly, Artificial Neural Network (ANN) needs a large dataset to learn, which probably takes a long training time and is not widely used for small datasets [22].

For small and imbalanced datasets, tree-based classifiers have proven to be one of the most efficient techniques [25]. Decision tree algorithms are one of the powerful supervisory machine learning techniques. They make decisions using bias and variance analysis to predict the labels. Furthermore, ensemble methods use the principle of combining weak learners to obtain a more reliable predictive model for better prediction and performance.

Ensembles can be obtained by boosting, which is a specific mechanism where learners gradually learn from the previous weak learners to reduce the overall loss function. This combined approach provides a powerful methodology for identification and pattern recognition for structured data [25]. XGBoost leverages the capabilities of boosting with ensembles. Moreover, we have identified that XGBoost is promising classifier amongst all the tree-based classifiers based on our preliminary results for ORNL dataset [15]. Furthermore, this method is not widely studied on power-grid based IDS applications. Consequently, we have decided to use the XGBoost model in RFE based feature selection scheme to obtain precise features. Further, it is also used as one of the classifiers to predict the output label in the majority vote ensemble method. We have listed pros and cons of each machine learning algorithms that we have used to generate majority vote based model in Table II.

IDSs for real-time systems such as SCADA-based power grids require low computational cost with high accuracy and execution speed. Such an IDS can be developed using a hybrid approach that combines the feature selection model along with an efficient classification scheme [26] which is the motivation for our proposed framework.

B. Ensemble Methods

A machine learning ensemble consists of a combination of several algorithms to obtain a result with better accuracy than from an individual classifier [27]. The ensemble is a statistical artifact known for over a hundred years based on the principle of *Wisdom of the Crowds* [28]. It was originally proposed by Sir Francis Galton who made a contest for observing a crowd in a cattle fair and showed that he was able to determine the weight of an ox by averaging the individual guesses from each

TABLE II
COMPARISON OF MACHINE LEARNING METHODS

Methods	Trees	k-NN	Artificial Neural Network	Naive Bayes
Description	Random Forest, XGBoost, Extra Trees and Adaboost are ensembles based on the decision trees	k-Nearest Neighbour looks for distance similarity between classes	Artificial Neural Networks are powerful classifiers able to work with complex patterns, able to represent non-linearity	Naive Bayes is a probabilistic classifier method
Characteristics	Decision trees, by bootstrapping (Random Forest) or by applying boosting (XGBoost, Ada Boosting) are easy to train and adaptable to different kinds of data	k-NN obtains good results when the data structure can be represented in a feasible representation space	consists of large number of "neuron" as processing elements, contains weighted connections to represent the distribution of data, acquire knowledge through learning process	Naive Bayes, based on Bayes theorem, has an assumption of feature independence, and requires prior probability estimates
Pros & Cons	Good fit for small datasets, Efficiently handled automatic step-wise feature selection, Does not required normalization and scaling	Effective for specific types, Needs homogeneous features, Flexible to choose the distance, No training is required, learn from neighbours	Expensive training for easy sets of data, but with complex data has superior ability to represent it	If probabilities are known obtains best results at low computational cost

person in a more precise way than the prediction of an individual.

There are three major classes of ensembles: bagging, boosting, and voting. Bagging and boosting use the same learner algorithm for prediction of the output labels. The difference between the two methods lies in how they generate successive subsets during classification. In boosting the datasets are randomly created, whereas in bagging, the elements are weighted, and not all of them have the same probability for selection [29], [30]. The third class, namely, stacking (voting) leverages several different algorithms working with the same data [31]. In a nutshell, bagging is used to decrease the model's variance; boosting works on the model's bias and voting achieves better performance by combining prediction of classification algorithms. The brief comparison of each of these three methods is listed in Table III.

TABLE III
COMPARISON OF ENSEMBLE METHODS: BOOSTING, BAGGING, AND STACKING

Method	Characteristics	Result
Bagging	homogeneous learners, parallel fitting, generate bootstrapping examples	focus on reducing variance Example : Random Forest
Boosting	homogeneous learners, sequential fitting, each time with wrongly classified samples	focus on reducing bias, model can be improved with gradient descent approach. Example : XGBoost
Stacking	heterogeneous learners, parallel fitting, meta-model combines learner results	focus on reducing bias Example : Majority Vote

In machine learning approaches, bagging is a powerful method to develop ensembles. The proposed method in [32] represents an example of a case study of neighbouring wind turbines based on bagging. This work was initially developed by Kramer *et al.* [33]. Bagging or bootstrapping aggregation consists of building independent predictors which extract the different samples from the training set and average the output by the prediction algorithms.

To achieve the best results, the predictors should be different or without correlation [27]. The voting ensemble creates multiple models and combines them to produce improved results. It is a more accurate classifier compared to the single predictive model.

Over a past few years, many Intrusion Detection Systems for various communication technologies have been proposed to detect the threats more accurately based on ensemble learning [34], [35], [36], [37], [38], [39], [40], [41].

One of the IDSs [34] is developed for imbalanced data samples (KDDcup99), where it is seen that J48 and Random Forest work best for big sample classes while others such as Bayesian network and Random tree seem to be a good fit for small samples. Therefore, the authors [34] propose a solution based on ensemble learning by applying a majority vote classifier to improve the performance of classification. Further, this work is improved by combining the prediction of Bagging and Boosting using ensemble techniques with tree base algorithms as the base classifier in [35].

In [36], the authors propose a novel approach that combines permission and intents supplements with an ensemble method for accurate malware detection for cellular phone communication. Moreover, in [37], authors execute anomaly detection over the communication networks by combining the prediction of three different types of classifiers, namely, neural networks, decision trees, and logistic regression using a weighted majority voting scheme.

The research work in [38] focuses on developing an IDS for network administrators by combining supervised and unsupervised learning techniques using ensemble method. This approach has been tested on various datasets like KDD Cup 99, NSK-KDD, and Kyoto 2006+ and is able to classify around 95% of the incoming traffic correctly [38]. In [39], the authors propose sustainable ensemble learning to improve the detection rate by aggregating multiclass regression models such that ensemble learning adapts to different attacks. Cloud-based solutions for distributed anomaly detection systems can be found in [40]. In [41], the authors propose a Gaussian

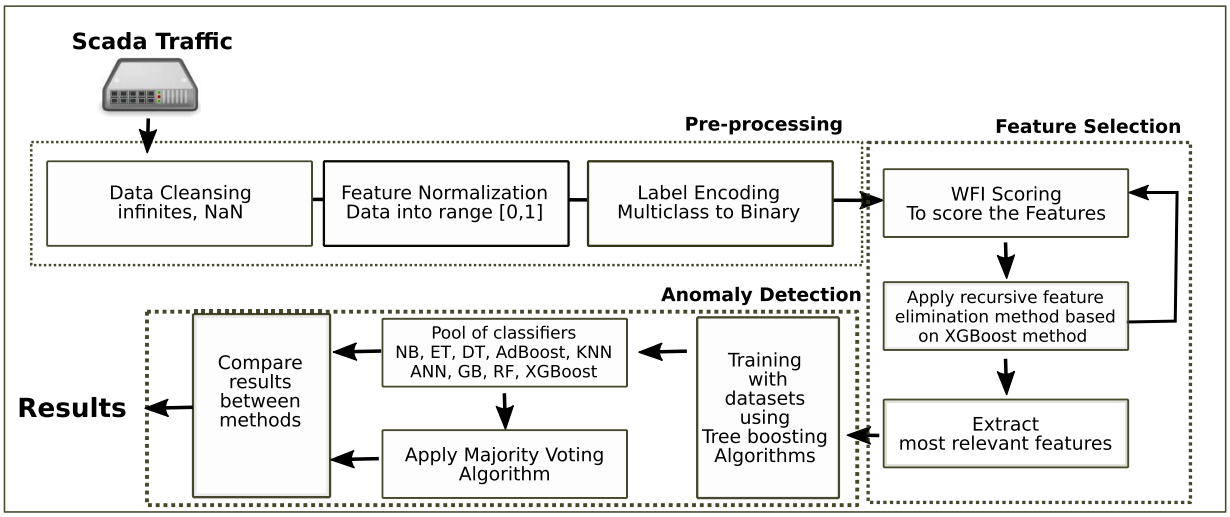


Fig. 2. Process diagram of the proposed framework for intrusion detection in power grids.

mixture based anomaly detection technique that relies on ensemble one-class statistical learning model that is designed to effectively recognize zero day attacks in real-time using the concept of edge networks.

III. Proposed Framework for Intrusion Detection System in Power Grids

This section presents the proposed scheme for an intrusion detection system for power grids to classify traffic into attacks and normal events by analyzing SCADA traffic. This novel approach uses the RFE-XGBoost based feature selection method to determine the most consistent features from the dataset based on feature importance scores. Furthermore, the majority vote ensemble method identifies accurate outcomes during classification. This combined approach accomplishes two significant aspects of real-time traffic monitoring namely, accuracy and computational speed. The entire framework is divided into three phases - data preprocessing, feature selection, and anomaly detection, as illustrated in Fig. 2.

The data cleaning, feature mapping, and feature normalization are done in the preprocessing phase to obtain streamed and sanitized data. Since the power grid is part of a large industrial control systems that use complex SCADA infrastructure to control the substation equipment, network monitoring devices such as SNORT, Wireshark and Syslog are used to obtain the different types of features from the communication data [16]. Usually, streaming data that is obtained from sensors or actuators in real-time systems has reliability issues, such as lost signal or wrong observations due to failures in measuring devices which result in their inability to interpret the scale readings. For this reason, the data cleansing operation is a critical process to remove incorrect data (like infinities or NaN data). In this phase we remove empty sequences that otherwise will generate issues such as inaccurate and faulty inferences with the algorithms. Moreover, the power grid records are collected at four PMUs (Phasor Measurement Units) which are situated at different locations in the power substation. Various internal attacks were launched by the ORNL to generate the IDS dataset for power grids reflecting the diverse nature of records. Another transformation that we performed in the data in this phase is the data normalization, to improve the training stability in the classifiers, especially for Artificial Neural Networks. For this normalization a standard scaler, a method that normalizes the records by considering zero mean and unit variance, was used.

In the feature selection phase, the importance of each feature is identified using the WFI scoring model. The recursive feature elimination approach is then applied to the binary dataset to eliminate irrelevant features recursively. Once the model determines the most consistent features, in the anomaly detection phase, the nine classifiers, namely NB, ET, DT, RF, GB, XGBoost, ADBoost, KNN, and ANN are used to predict the output labels. Finally, the majority vote-based ensemble method predicts the class label for input samples based on the majority of the class labels predicted by each of these nine classifiers. The voting classifier uses “hard voting” to classify the input sample based on the majority class label.

IV. Concept of Methodology

A. Majority Voting Algorithm

There are two main categories of majority-based ensemble methods, namely, voting and averaging [42]. Generally, voting is used for classification, while averaging is used for regression. We have used a voting based ensemble method to detect intrusions. In this method, we can create multiple base models using a training dataset. The output of each base model acts as the input of the majority vote base ensemble algorithm. These base models are created using different splits of the same training dataset along with other classifiers. The majority vote classifier predicts the output label based on the prediction of multiple base models. To calculate the overall error, we assume that the probability of each base model being correct is $(1-\epsilon)$, where ϵ is the classifier error. We assume that the

Algorithm 1. Majority vote ensemble training algorithm for n classifiers

Data:
Dataset **Train** $= \langle \mathbf{X}, \widehat{\mathbf{Y}} \rangle$, **Test** $= \langle \mathbf{x}, \widehat{\mathbf{y}} \rangle$,
Size of Test Dataset: m
Classifiers $\mathbf{C} = \langle c_i | i \in 1 \cdots n \rangle$
begin
  for$i : 1$ **to** n **do**
    $p_i \leftarrow$ train predictor (c_i) on Test Dataset
  end
  for $i \leftarrow 1$ **to** m **do**
    for $j \leftarrow 1$ **to** n **do**
      Apply predictor(c_j) to sample x_i
    end
    best prediction = more classifier votes
    $\hat{y} \leftarrow$ best prediction
  end
end
Result:
Predictions: $\widehat{\mathbf{y}} = \langle y_i | i \in 1 \cdots m \rangle$

classification errors are independent, and we can also obtain the probability of the majority vote error by applying binomial distribution. The probability of obtaining k valid predictions out of n (k over 50% or $k > n/2$) is achieved using binomial distribution as follows:

$$\text{Probability}(X = k) = \binom{n}{k} \epsilon^k (1 - \epsilon)^{n-k} \tag{1}$$

We obtain the total probability by adding all the individual probabilities for each k:

$$\text{Total Probability} = \sum_{k > \frac{n}{2}}^{n} \binom{n}{k} \epsilon^k (1 - \epsilon)^{n-k} \tag{2}$$

If $\epsilon < \frac{1}{2}$, and the predictions from the classifier are considered as independent, the error is, in principle, smaller, as when $n \to \infty, \epsilon \to 0$. With the majority vote strategy, we can obtain better accuracy than with the direct or linear-averaged approach. The majority vote model gives the same weight to each one of the votes using a *democratic approach* (see Algorithm 1)

If we observe that some inputs are more potent than others, then we can quantify and adjust this contribution. For instance, with a Bayesian model averaging (BMA) where the weighting is adjusted after training by reviewing the individual contributions to accuracy one by one.

The other type of ensemble approach are short term algorithms where ensembles are applied for short term energy demand forecasting [43], [44]. The use of ensembles combined with deep or machine learning algorithms is a promising area of research as the ability to run multiple algorithms in parallel is efficient, and the combination of models with different strengths generates better results.

B. Feature Selection

Through feature selection, we can select the subset of relevant features for the appropriate model construction. This will avoid the bane of dimensionality and enhances the generalization of the model by reducing overfitting [45]. However, due to this approach, some feature information may be lost, but that does not impact the overall performance of the model; instead, the selected features are more representative to model the classifier. Moreover, the samples with hundreds of features will increase the computation cost and decrease the classification performance. Therefore, our first target is to identify the subset of the relevant features of the dataset which are highly related to the class but are not related to each other.

To identify the most consistent features of power grid datasets, we have used the WFI based scoring model by ranking the elements. This method extracts the feature importance score of each feature by considering the improvement in impurity while splitting the individual tree. The irrelevant features are removed recursively according to the scoring model using the RFE approach on XGBoost algorithm. We have used binary datasets to extract the most relevant features instead of multi-class datasets, as sometimes the WFI scoring model has a bias towards multiple categories of the dataset [46]. However, the extracted features are applicable to all four categories of the datasets.The features are carefully removed without losing much of the information to generate the feature subset using RFE approach. In each iteration, XGBoost is trained with a selected feature subset to measure the accuracy.

During the process of feature selection, the current subset is replaced by the selected set of features when the accuracy of the current subset is increased by more than 0.5%. This way, we can achieve consistent elements from the entire dataset. The steps of the RFE-XGBoost algorithm are shown in Algorithm 2.

V. Experiments and Results

A. Datasets

To determine the performance of the proposed approach, we have used three public benchmark datasets [16]. These datasets were created at Oak Ridge National Laboratories (ORNL) by setting up a power grid testbed [10]. This testbed was configured using various power grid components, namely, power generators - G1 and G2, IEDs - R1 to R4, breakers - BR1 to BR4 and a three-bus two-line transmission system. In the case of fault detection, the IED trips the corresponding breaker depending upon the nature of the fault. However, these IEDs are not smart enough to differentiate between original and fake failures. Moreover, operators can also manually trip the breakers and other system components during system maintenance [9].

The datasets derived from this power grid testbed contain measurement related to normal, disturbance, control, and cyber-attack behaviors captured during electrical transmission [11]. These datasets are randomly sampled and classified into three main categories, namely, binary, three state, and multi state. Initially, the multi state dataset is constructed during the experiment, and consists of a total of 37 scenarios. These scenarios are mainly divided into three categories, namely, 8 natural events, one no event, and

Algorithm 2. Recursive Feature Elimination based on XGboost WFI scoring model

Data:
Training power-grid data-set: PD
begin
Initialize:
Current features $Curr\_PD$ = {1,2,3,...,n}
Ranked features $Sel\_PD = Curr\_PD$
Set standard deviation SD = 0.5
Set proportion of features to be deleted = $SetProp$
Build XGBoost model based on $Curr\_PD$
Compute the initial accuracy $Acc(Curr\_PD)$
while *Features(Curr_PD) != Empty* **do**
Evaluate the ranking criteria
Rank features of $Curr\_PD$ in ascending order by WFI scoring model
Remove features = SetProp(min($Score$))
Store the features in Sel_PD
Build $XGBoost$ model based on the rank features Sel_PD
Compute the accuracy $Acc(Sel\_PD)$
if $Acc(Curr\_PD) + SD < Acc(Sel\_PD)$ **then**
$Curr\_PD \leftarrow Sel\_PD$
else if $Acc(Curr\_PD) == Acc(Sel\_PD)$ & $Ftr(Curr\_PD) > Ftr(Sel\_PD)$**then**
$Curr\_PD \leftarrow Sel\_PD$
end
$Best\_PD \leftarrow Cur\_PD$
end
Result: ranked feature subset $Best\_D$

28 attack events. The eight natural events are further divided into 6 SLG faults events and 2 line maintenance events, as listed in Table IV. Moreover, the 28 attack events are subcategorized into three major attack events, namely, Data Injection, Remote Tripping Command Injection, and Attack on Relay Settings. These include 6 SLG fault replay attacks, 4 command injection attacks against single IED, 2 command injection attacks against 2 IEDs, 10 relay setting change attacks on a single IED, 4 relay setting change attacks on 2 IEDs, and 2 relay disable and line maintenance attacks as listed in Table IV. These attack scenarios are simulated using the concept of an internal intruder, who can launch different attacks by issuing malicious injections from the substation [10]. Moreover, we have derived a seven-states dataset from the multi-states dataset. The dataset of each category is sub-sampled into fifteen sets. Table IV gives the summary of various output labels according to the four categories of the dataset.

The datasets of the power grids consist of a total of 128 features. These features are derived using 4 Phasor Measurement Units (PMUs), which measure electrical signals of substation using a common time source for effective time synchronization. A total of 106 PMU measurements are carried out using 4 PMUs, where each PMU measures 29 features of a particular location. These features are referred to as R# (signal Reference), which indicate the index of PMU and type of measurement. For example, R2-PA2: IH represents the phase A - current phase angle measured by PMU located at R2 [16]. Twelve different categories indicate phase angles and magnitude of voltage and

TABLE IV
DESCRIPTION OF THE OUTPUT LABELS OF THE VARIOUS CATEGORIES OF THE DATASETS

Categories	Output Labels
Binary (2)	Normal, Attack
Three States (3)	Normal, Attack, No Event
Seven States (7)	1-Natural SLG Fault, 1-Data injection attack 2- Remote Tripping Command injection attack 3- relay setting change attacks
Multi States (37)	1 – No event 8 – Natural events - 6 SLG faults - 2 Line maintenance events 28 – Attack events - 6 data injection SLG fault replay attacks - 4 command injection attacks against single IED - 2 command injection attacks against two IEDs - 10 relay setting change attacks on single IED - 4 relay setting change attacks on two IEDs - 2 relay disable and line maintenance attacks

current. The detailed description of the features is given in [15]. Furthermore, 16 more features are derived using control panel logs, snort alerts, and relay logs [10]. The last column refers to a marker that labels different normal and malicious events. Each set consists of around 5000 instances that include 294 no events, 1221 natural events, and 3711 attack events approximately, which represents that the given datasets are imbalance in nature.

B. Evaluation Methodology

The primary objective of the proposed model is to provide a real-time intrusion detection for power-grid systems. Hence, our target is to build a fast and accurate model that captures any malicious event efficiently that may happen in the network. To fulfill both requirements, we have used RFE-XGBoost-based WFI scoring model for feature selection along with the majority vote-based ensemble method for classification. The feature selection module improves the computational cost as we are targeting the 30 most consistent features out of 128 features of the given datasets. Furthermore, we have used nine most powerful classifiers to classify the normal and malicious events. For more accurate results, we have applied the majority vote-based ensemble method, which predicts the class label based on majority of the class labels predicted by each of these classifiers.

These datasets used in our analysis are the publicly available datasets generated at the ORNL laboratory on a small power grid testbed [16]. For proper validation, experiments were computed for four different categories of the samples. Furthermore, the observations were carried out using 100,000 normal and attack events of each of these four categories, which were divided into 15 datasets. For fair distribution and assessment, each dataset was split randomly into two subsets, training (80%) and testing (20%). The training data was used for the algorithm training and the testing data was used to test the accuracy of the result. To avoid selection bias of the datasets and to reduce the overfitting, we have used 10-fold cross-

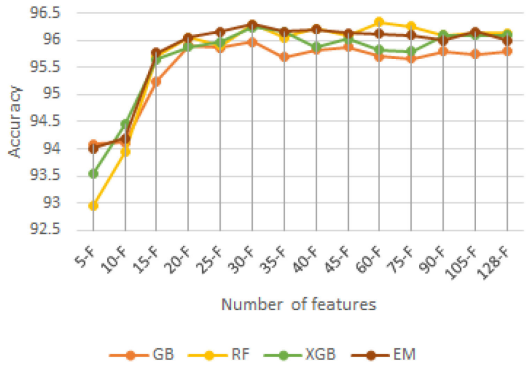


Fig. 3. Comparative analysis of different features to evaluate the accuracy using RFE-XGBoost WFI scoring model.

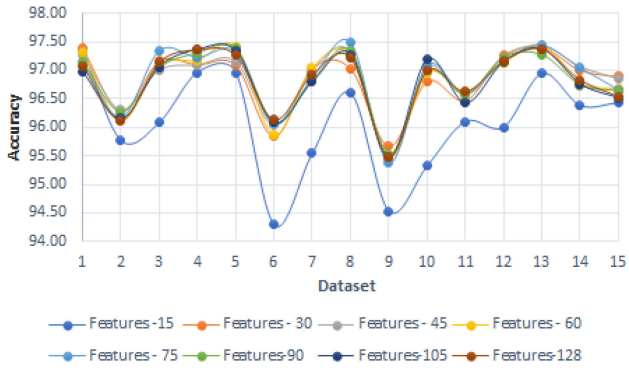


Fig. 4. Different number of features are evaluated to measure the accuracy of 15 binary datasets using RFE-XgBoost WFI scoring model.

validation technique during the training process. This method performs the training 10 times with different random selections (80/20) from the original dataset. This well-defined systematic approach circumvents the inadequacy of bias performance assessment. The proposed approach is implemented using Python on a Jupyter notebook using the Anaconda distribution platform on Windows10 with Intel Core i5-8300H 2.30GHz processor, 8 GB RAM, and Nvidia Geforce GTX 1060 GPU.

C. Evaluation of Feature Selection

We have made observations based on the number of subsets of the features considering 15 binary datasets. Initially, we started with 128 features and reduced the number of features in each iteration based on the output of the WFI scoring model to compare the accuracy of the current set with the selected subset. To extract the gist of the features, we have applied WFI based scoring model which scores the importance of all features. This ranking defines how often the feature is used to determine the output label while constructing tree.

Fig. 3 illustrates the comparative analysis of different features versus accuracy graph of one of the 15 datasets. The classification with 30 features offers the highest accuracy during classification of normal and attack events using the majority vote-based ensemble classifier.

Fig. 4 demonstrates the accuracy of different 15 datasets according to the various subsets of the total features. The accuracy increased significantly up to 30 features, after that there is no substantial improvement in accuracy. Hence we have extracted most 30 features of each dataset and consider the same features for all the four categories, namely, binary, three-class, seven-class and multi-class datasets.

D. Result Discussion

To evaluate the performance of the majority vote based ensemble algorithm, we have computed the accuracy of fifteen datasets of all the four categories using nine most promising classifiers. The choice of these classifiers is carried out based on our preliminary results of the comparative analysis of various machine learning classifiers [15]. We have chosen nine heterogeneous classifiers to determine the efficiency of selected features via multiple simulation trials and observed the predictions of all the algorithms. After deriving the accuracy of all the nine classifiers, the majority vote ensemble algorithm was applied to compare the prediction of the output labels. The comparison was carried out based on the majority class label voting classifier with "hard voting" to classify the input samples.

The ensemble algorithm predicts accurate outcomes by aggregating and applying the majority vote rule on the result of the different classifiers. We have incorporated heterogeneous classifiers, namely, random forest (RF), gradient boosting (GB), XGBoost (XGB), Extra Tree (ET), Decision Tree (DT), K-Nearest Neightbour (KNN), Naive Bayes (NB), Adaboost - Decision Tree (AdBoost-DT), and artificial neural network (ANN) to achieve performance improvement of the majority vote based ensemble model.

We have performed overall 60 computations of each of the four categories (binary, three-states, seven-states, and multi-states) containing fifteen datasets to evaluate the performance of each of ten classifiers. According to the analysis, the accuracy of the Naive Bayes algorithm is less compared with other classifiers for all the four categories, namely, binary (around 52.34%), three class (58.21%), seven class (19.26%), and multi class (13.2%). Fig. 5 presents a comparative analysis of the accuracy of the remaining eight classifiers along with the majority vote-based ensemble algorithm. Among nine base classifiers random forest, gradient boosting and XGBoost have mostly proven to be more efficient in the case of binary, three states, and seven states classification. However, for multi states classification, random forest, extra tree and XGBoost are more promising than the other six classifiers.

Moreover, the majority vote ensemble classifier outperforms by taking advantage of prediction logic of other nine classifiers. The accuracy of the majority vote based ensemble method is higher and more precise than the other nine classifiers with accuracy around 98.24% for binary, 97.95% for three states, 95.91% for seven states, and 93.78% for multi states datasets, approximately.

To validate the effectiveness of the proposed scheme, we have compared the accuracy of majority vote based ensemble algorithm with five published methods, namely AdaBoost-JRIP (AdaJRIP) [9], Common Path Mining [10], [11],

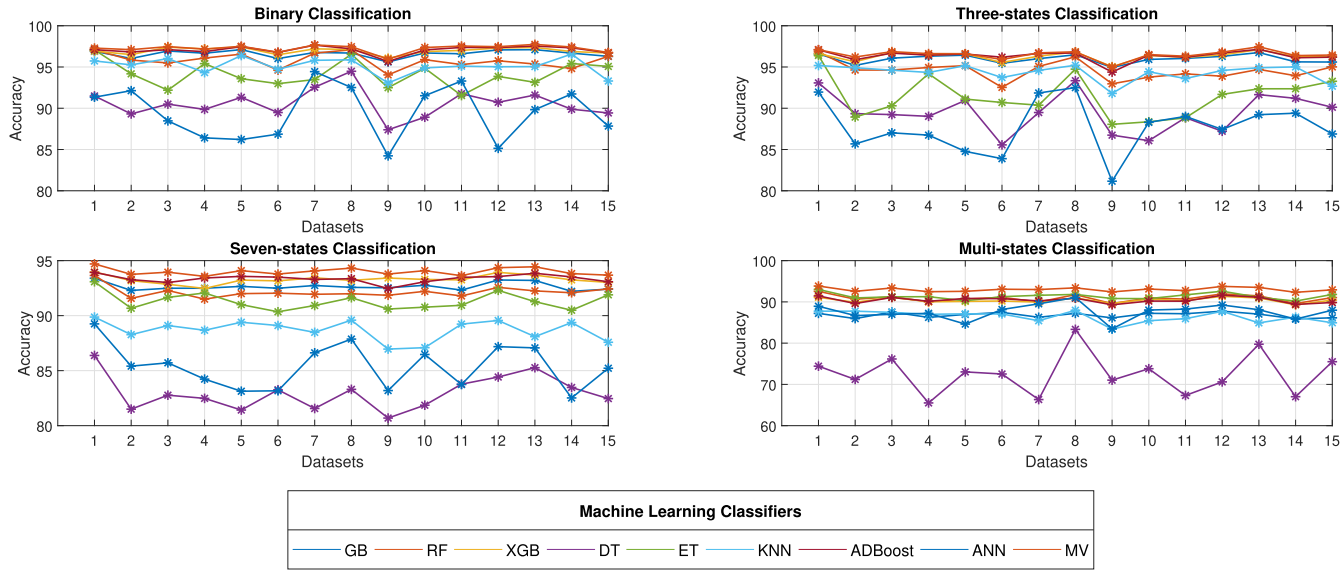


Fig. 5. Comparative view of different Machine Learning classifiers for four categories for each of the fifteen datasets.

TABLE V
COMPARATIVE ANALYSIS OF OVERALL PERFORMANCE OF VARIOUS TECHNIQUES AND PROPOSED MAJORITY VOTE ENSEMBLE METHOD BASED CLASSIFIER

Classifier	Data Cleaning	Feature Selection	Features (%)	Classes	Accuracy
ADA-JRIP [9]	NA	NA	100%	2	94.55%
				3	94.61%
				37	85.85%
CPM [10],[11]	Applied	NA	100%	7	93.00%
				25	90.40%
EMCT [12]	Applied	PCC	25%	2	70.60%
			50%		76.3%
			75%		83.5%
			100%		90.2%
GMMKM [14]	Applied	PCC	25%	2	94.56%
			50%		95.83%
			75%		96.82%
			100%		97.27%
Tree Based [15]	Applied	GBFS	12%	2	97.26%
				3	96.50%
				7	94.12%
				37	92.46%
MV-EM	Applied	RFE-XG	**25%**	**2**	**98.24%**
				3	**97.95%**
				7	**95.91%**
				37	**93.78%**

Expectation Maximization Clustering Technique (EMCT) [12], Gaussian Mixture - Kalman Filter Model (GMM-KF) using Pearson Correlation Coefficient (PCC) feature selection method [14] and GBFS based tree based classifiers [15].

Furthermore, we have also compared various performance evaluation factors such as whether proper pre-processing is applied on datasets; whether feature selection approach is incorporated and if applied how many features are selected to evaluate the accuracy by considering four states of dataset. Table V shows that the proposed framework outperforms compared to other published techniques. The model accomplishes significant accuracy for all the four categories by selecting only 25% of the features. Note that the results mentioned in the table refer to the highest accuracy achieved during the classification by the majority vote based ensemble algorithm.

TABLE VI
COMPARISON OF ACCURACY OF DIFFERENT ENSEMBLE TECHNIQUES

Categories	Bagging ensembles			Boosting ensembles			Stacking
Classifiers	**DT**	**RF**	**ET**	**GB**	**AdB(DT)**	**XGB**	**Majority**
Binary	91.32	96.08	96.54	96.76	95.21	96.93	98.24
Three-Class	93.07	96.84	96.4	96.59	95.26	96.61	97.95
Seven Class	86.38	93.65	93.08	93.4	92.16	93.97	95.91
Multi-Class	74.42	92.56	93.02	87.22	91.59	91.1	93.78

Bagging generally considers homogeneous weak learners to train the model sequentially. However, the learning process occurs independently, and prediction is made by averaging all the parallel models. On the other hand, boosting learns sequentially by considering errors from previous ones. In both these methods, homogeneous learners are used. In contrast, stacking often considers heterogeneous weak learners to train the meta-model to predict the output based on different model predictions. We have discussed the literature pertaining to various ensemble methods in Section II, namely, bagging, boosting and stacking. To demonstrate the efficiency of our proposed approach, we have compared bagging and boosting based ensemble methods with the majority vote based ensemble technique which refers to stacking approach.

As shown in Table VI, bagged DT, RF and ET are examples of bagging ensembles whereas boosting ensembles include GB, AdB-DT and XGB. Furthermore, we have designed the majority vote ensemble technique by applying the predictions of nine heterogeneous classifiers. Table VI represents the promising results compared to other ensemble techniques in terms of accuracy. Moreover, the other three non-ensemble classifiers, namely, NB, KNN and ANN have less accuracy; 54.29%, 94.32%, 88.47% for binary, 58.21%, 93.72%, 87.02% for three state, 21.57%, 89.10%, 84.23% for seven

TABLE VII
SPECIFICATION OF EACH MODEL ARCHITECTURE

Model	Main Parameters
Gradient Boosting	estimators = 100, max depth = 12, min split = 12
Random Forest	estimators = 100, criterion ='gini', max depth = 12, min samples leaf = 1, min split = 2
XGBoost	estimators = 100, max depth = 12, random state = 3
Decission Tree	max depth = 12, min split = 2, random state = 3
Extra Tree	estimators = 100, max depth = 12, min split = 2
AdaBoost	classifier = DT, max depth = 12, min split = 2
ANN	dense = 512, epochs=50 batch=16, opt=adam, act=relu
KNN	neighbours = 3

state, and 13.18%, 87.77%, 83.13% for multi state as compared to the majority vote ensemble method.

We have denoted the specification of each model in Table VII. These parameters are achieved using a grid search while training the model for hyper-parameter tuning, which improves the efficiency of each classifier. Here, estimators refer to the number of trees created in the model during the training process. At the same time, maximum depth (max depth) represents the node expansion until all leaves contain less than the value defined in the minimum samples split (min split). For the ANN model, we have created 512 hidden layers with 50 epochs each by considering a batch size equal to 16. Furthermore, we have considered 'adam' optimizer for weight optimization and 'relu' as activation function for the hidden layer. KNN decides the output label by considering the prediction of three nearest neighbors.

Using an ROC plot, we can visualize the trade-offs between the true positive rate (TPR) also known as sensitivity and false positive rate (FPR). Further, the Area Under the Curve (AUC) presents the degree of separability, which defines the capability to differentiate the classes. Fig. 6 shows the ROC curves of four classifiers created by the 10-fold cross-validation method. We have examined RF, GB, XGBoost and Majority Vote, which represent the different categories of ensemble technique, namely, bagging, boosting, and voting ensembles.

Moreover, we have presented the ROC curve of one of the fifteen datasets of each of the four categories. ROC curve qualifies the model according to the total area under the curve for each classifier. The metric falls between 0 and 1, with a higher value indicates better classification performance. The graphs in Fig. 6 compare the AUC of four classifiers. The green curve represents the majority vote-based ensemble method is contributing to the high AUC scores for all four classes. This means that the majority vote based model is better at achieving a blend of precision and recall. Furthermore, random forest and XGBoost contribute slightly better than gradient boosting for all four categories. However, gradient boosting is comparatively lower in terms of AUC scores specifically for the multi states dataset.

In the case of imbalanced datasets, the PR plot is more informative than the ROC plot while evaluating classifiers [47]. Here we are not only targeting binary classification but also classifying multiple attack events. Hence, for more information retrieval, we have also analyzed PR curves in case of bias in the class distribution. The baseline of the PR curve is determined

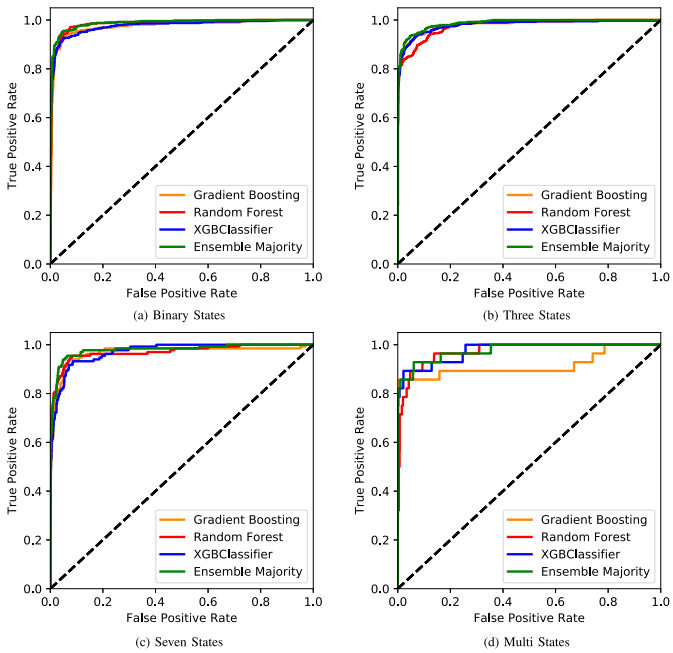
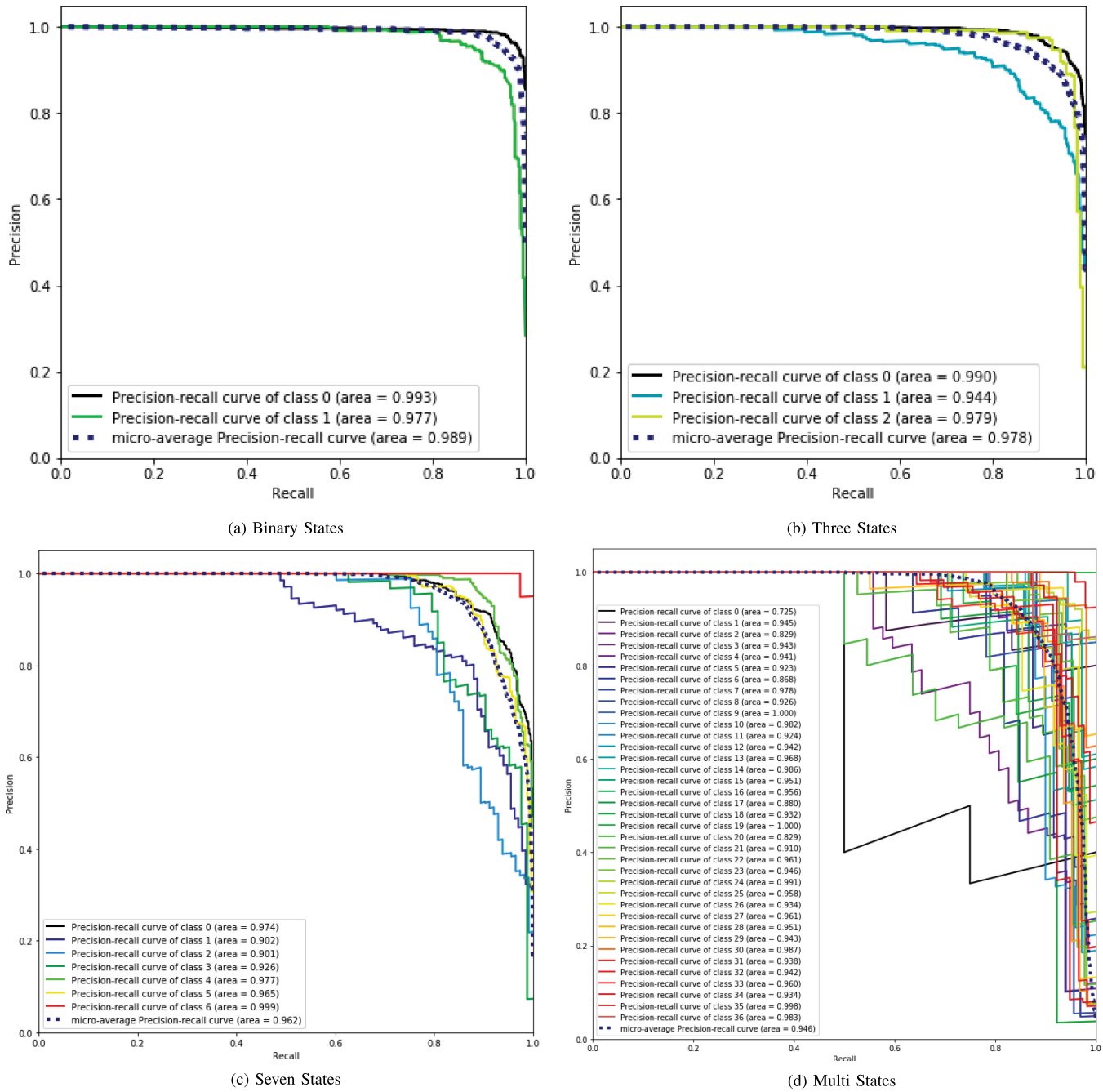


Fig. 6. ROC Curves of three types (bagging, boosting and stacking) of Machine Learning Classifiers for four categories.

by the relation of precision and recall values. Fig. 7 depicts the precision/recall for each threshold for a majority rule-based ensemble model by considering all the four categories of the dataset. For all the four types, the majority rule-based ensemble classifier maintains a high detection rate. The proposed model has achieved 98.9%, 97.8%, 96.2%, and 94.6% of the average precision-recall curve area for binary, three states, seven states, and multi-states, respectively. The exact percentage of each output label is depicted in Fig. 7. The results indicate the model performs exceptionally well with all the categories to predict various types of class labels.

Precision defines the ratio of the number of true positives, divided by the total number of true positives and false positives, which describes the efficiency of the model in terms of prediction of the positive class. Recall represents the ratio of the number of true positives divided by the total number of true positives and false negatives. While F measure is used to combined the precision and recall to determine the harmonic mean of those parameters. For the precise assessment, we have measured the efficiency of our proposed model not only by evaluating the accuracy of the classification but also by considering other factors such as, recall, precision, F1 score and miss rate. We have achieved high precision, recall, and F measure for RFE based majority vote ensemble method for all the four categories. The results of these performance metrics are illustrated in Table VIII. We have evaluated the results for all the 15 datasets of all four categories. However, we have depicted the most promising result of all the observations in Table VIII. Furthermore, for a more detailed view, we have represented the results of all the simulation trials in Fig. 8, which consist of all the 15 datasets of binary, three states, seven states, and multi states categories. As shown in the figure, we have achieved around 97% detection rate, which offers significant classification of



(a) Binary States (b) Three States

(c) Seven States (d) Multi States

Fig. 7. Precision-Recall Curves of RFE based Majority vote ensemble method for four categories.

attack and normal events for binary and three states categories, with only 3% miss rate. Furthermore, the seven class and multi-class output labels are also accomplished with 93% detection rate with around 7% miss rate.

We have observed the importance of the various features in the previous section, where accuracy is measured by considering subsets of the features. In that, we have focused on the binary dataset. For further proof of concept, we have evaluated the accuracy of three other categories, namely, three class, seven class, and multi class datasets, by comparing all the 128 with 30 features. To extract the gist of the features, we have applied an RFE based WFI scoring model, which scores the importance of all features recursively. This ranking defines how often the feature is used to determine the output label while constructing the tree. Table IX illustrates the comparative analysis of four categories by considering 128 features

TABLE VIII
PERFORMANCE EVALUATION METRICS OF PROPOSED RECURSIVE FEATURE ELIMINATION BASED MAJORITY VOTE ENSEMBLE METHOD

Measure	Binary	Three-class	Seven-Class	Multi-class
Precision	97.40%	97.21%	95.38%	94.04%
Recall	96.63%	96.1%	93.46%	92.79%
F-Measure	96.99%	96.6%	94.26%	93.04%
Miss Rate	3.37%	3.90%	6.54%	7.21%

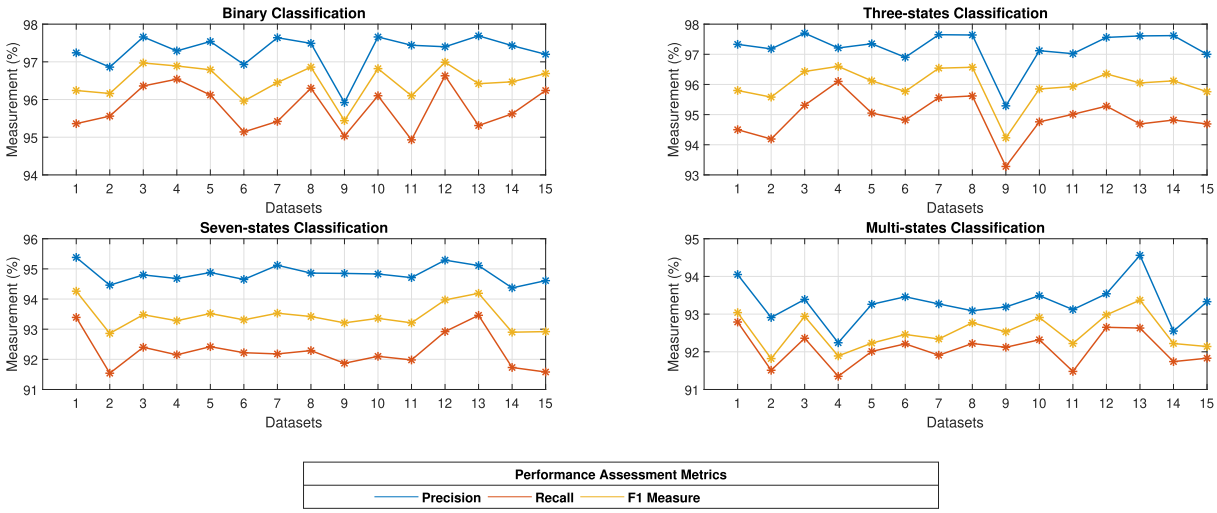


Fig. 8. Result of various performance measurements (Precision, Recall, F1 Measure) of RFE based majority vote ensemble method for four categories of fifteen datasets.

TABLE IX
COMPARISON OF ACCURACY OF MAJORITY VOTE ENSEMBLE ALGORITHM WITH AND WITHOUT RECURSIVE FEATURE ELIMINATION BASED FEATURE SELECTION

Classifiers	**Without Feature Selection** 128 features (Accuracy)	**RFE - Feature Selection** 30 features (Accuracy)
Binary	96.93	97.44
Three-Class	96.64	97.25
Seven Class	93.65	94.91
Multi-Class	92.23	93.08

TABLE X
EXECUTION TIME OF RANDOM SAMPLE OF 5300 RECORDS

Phase	Execution time (secs)
Pre-Processing time	0.0186300039
Feature Selection time	1.646741629
Training Time	306.036

TABLE XI
COMPARATIVE ANALYSIS OF TRAINING TIME OF VARIOUS CLASSIFIERS (RANDOM SAMPLE OF 5300 RECORDS)

Various Classifiers	Training time (sec)
Gradient Boosting	101.2503724
Random Forest	9.819750547
XGBoost	7.018202782
Decision Tree	0.892643929
Extra Tree	3.217407703
K-nearest neighbors	3.426834106
Naive Bayes	0.041889906
AdaBoost (DT)	42.29490685
Artificial neural network	240.9818392
Majority Vote	306.0361404

versus 30 features extracted by RFE. The classification with 30 features offers the highest accuracy during the classification of normal and attack events using the majority vote ensemble classifier. In Table IX, we have presented the result of one of the 15 datasets. During experiments, we have also observed that the training time of multi states datasets with all the 128 features is unrealistic as it took more than 24 hours. Hence, feature selection is a crucial factor used to develop a better predictive model and make the model computationally efficient.

The detection time is determined using real-time data classification based on incoming traffic (generally based on one observation). Intrusion detection systems should provide an immediate response to potential attacks. To improve the performance of such systems we need to eventually train the module based on the behavior of real-time traffic and accordingly need to deploy the model in a real-time environment. Since training involves computational time and resources, it is mostly performed using high-performance infrastructure (generally offline on the plant floor or at the control center). In contrast, the intrusion detection inference engine (trained model) is used to classify the observation of real-time traffic and deployed in hardware that is connected to the communication network.

We have conducted experiments to determine the execution time of each of the four phases, namely, pre-processing, feature selection, training time, and testing time of the proposed technique by taking random samples from the original dataset (5300 records out of 100,000 records). The execution times reported refer to the implementation of the proposed approach on Windows10 with Intel Core i5-8300H 2.30GHz processor, 8 GB RAM, and Nvidia Geforce GTX 1060 GPU. The execution times of all the modules of the proposed algorithm are listed in Table X.

The above refers to the training time of the three phases. Generally, the preprocessing, feature selection, and training of the model are performed frequently at certain time intervals at the plant floor/control center offline using high computational resources. However, to address the detection rate of the proposed scheme we need to target real-time classification. In principle, the filtering mechanism of the proposed algorithm should be incorporated in edge computing devices such as smart routers and smart switches. This will significantly reduce the detection rate as filtering (preprocessing) is

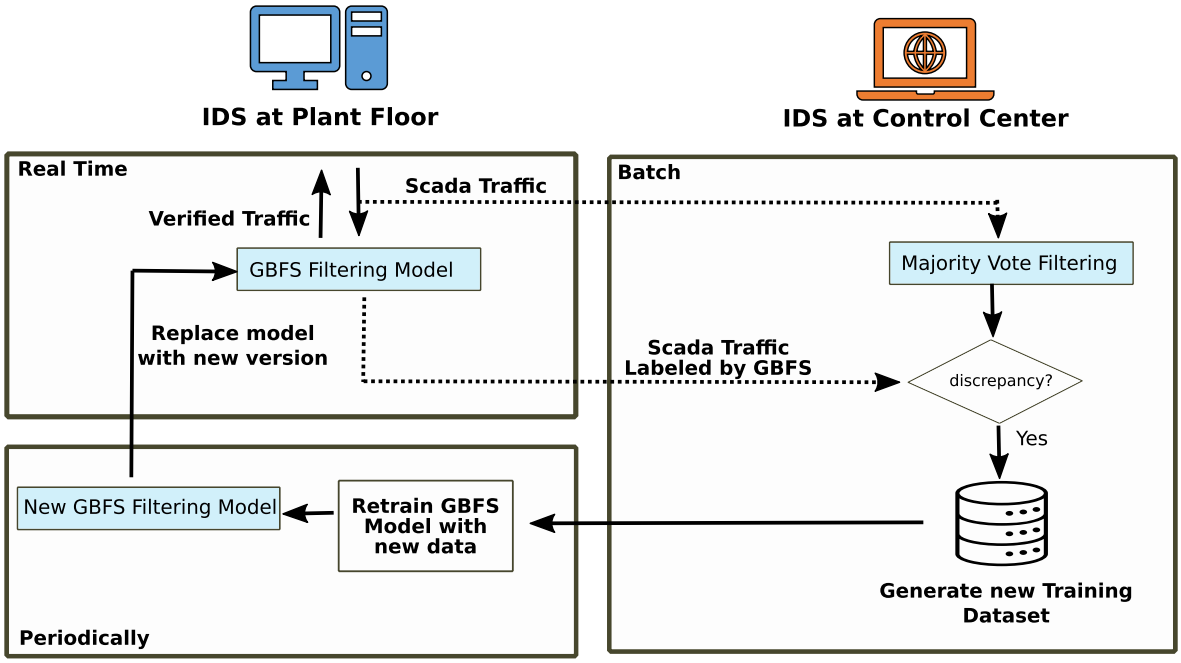


Fig. 9. IDS framework for real-time SCADA systems for power grids.

computed on hardware. These distributed computing devices make the detection rate low (nanoseconds) which also avoids the requirement of a powerful CPU or the support of a GPU. Thus, the execution time to classify normal/attack events by our proposed model is comparatively low which is adequate for a real-time intrusion detection system.

Furthermore, we have compared the training time of various classifiers with the majority vote-based ensemble method as depicted in Table XI. While the majority vote ensemble method takes more time compared to the single classifier, to balance the execution time, and to obtain high performance, we have proposed a real-time IDS for SCADA systems as discussed in Section VI. In particular, we have deployed a majority vote-based IDS on the control center that monitors all the plant floor IDSs which work on a single GBFS based classifier. This approach maintains the performance of IDS for real-time SCADA systems to distinguish attacks and normal events during live data streaming with the standard available hardware.

VI. Proposed IDS Framework for Power Grid SCADA System

We have extended our previous GBFS based model with RFE based majority vote ensemble method by combining the results of several classifiers to achieve an accurate outcome. The purpose of the previous model is to achieve accurate classification without deteriorating the performance of the system using prediction of a single classifier. However, majority vote ensemble method predicts the output label based on the majority of the output labels predicted by each classifier. This will further improve the efficiency of the prediction and provides the most accurate output label in terms of normal and attack events. For that, we have targeted various heterogeneous classifiers, namely, Random Forest, Gradient Boosting, XGBoost, Artificial Neural Network, Naïve Base, and Decision Table for ensemble learning by referring to preliminary results from this paper [15]. This approach will generate a better predicting model than a single model using a hard voting based majority rule ensemble technique.

In distributive environments such as power grids, the availability of the most accurate intrusion detection system is a crucial factor. This is achieved by replacing the existing deployed model with the most recent ones, which enhances the capability of IDS and is accomplished by training the model frequently according to the live traffic. The training time plays a significant role in real time detection as shorter execution time develops the model quickly. We have proposed the IDS framework for real-time SCADA systems for power grids, as shown in Fig. 9. In this approach, we place two different IDSs at two different locations, one at the plant floor and another at the control center. The plant floor IDS analyzes the SCADA traffic using the GBFS based filtering model as it is more compatible in detecting the intrusions in real-time communication. However, for more accurate results, the output of this module is verified at the control center using the majority vote-based IDS with multiple classifiers. In case of a discrepancy in the output labels, the records will be added to a new training dataset to retrain the GBFS filtering model periodically. This way, we can achieve the most updated test model and replace the existing model with the recent model. Through this approach the proposed framework achieves high computational speed and accurate prediction for live SCADA traffic of power grids.

VII. Conclusions

This paper presents a RFE- XGBoost based feature selection approach along with the majority vote-based ensemble method for intrusion detection in power grids. The proposed framework comprises of three key elements, namely, data preprocessing, feature selection, and anomaly detection. Initially, during data preprocessing, the features are mapped and scaled to a specific range. The RFE-XGBoost based feature selection approach is subsequently applied on filtered data to compute the most stable features from the entire dataset. This approach enhances the learning efficiency. Furthermore, the selection of the features is carried out dynamically according to network traffic. In the subsequent stage, these reconstructed datasets are used by nine heterogeneous classifiers to predict the various attacks and normal events. Finally, the majority vote-based ensemble algorithm is applied to predict the output based on the majority of the class labels predicted by each of the nine classifiers.

The experimental results reveal that the proposed framework fares well in terms of accuracy, detection rate, precision, and recall. Moreover, the proposed model outperforms some of the state-of-the-art published techniques. The model offers a blend of effectiveness with precision, as it uses the limited number of stable features, and the classification is carried out based on combined predictions of nine most promising classifiers. Moreover, this combination requires limited computational cost, which is one of the crucial factors for mission-critical applications. Thus the proposed model has the potential to leverage the competencies of real-time SCADA systems for power grids.

Acknowledgments

Thanks to Brian Stacey and Rohit Joshi of Cistel Technologies for their valuable feedback.

References

[1] B. Krebs, "Cyber incident blamed for nuclear power plant shutdown," Washington Post. [Online]. Available: http://www.washingtonpost.com/wp-dyn/content/article/2008/06/05

[2] B. Kesler, "The vulnerability of nuclear facilities to cyber attack," *Strategic Insights*, vol. 10, no. 1, pp. 15–25, 2011.

[3] H. Xu, Y. Lin, X. Zhang, and F. Wang, "Power system parameter attack for financial profits in electricity markets," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3438–3446, Jul. 2020.

[4] "SANS and Electricity Information Sharing and Analysis Center (e-isac). analysis of the cyber attack on the ukrainian power grid," Accessed: Sept. 28, 2019. [Online]. Available: http:// www.nerc.com/pa/CI/ESISAC/Documents/E-ISAC_SANS_Ukraine_DUC_18Mar2016.pdf

[5] K. Poulsen, "Feature importance of feature selection," 2003. Accessed: Sept. 10, 2019. [Online]. Available: https://www.securityfocus.com/news/6767

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

[7] D. Upadhyay and S. Sampalli, "Scada (supervisory control and data acquisition) systems: Vulnerability assessment and security recommendations," *Comput. Secur.*, vol. 89, 2020, Art. no. 101666.

[8] B. Sussman, "Revealed: Details of 'first of its kind' disruptive power grid attack," Accessed: Mar. 21, 2020. [Online]. Available: https://www.secureworldexpo.com/industry-news/first-u.s.-power-grid-attack-details

[9] R. C. Borges Hink *et al.*, "Machine learning for power system disturbance and cyber-attack discrimination," in *Proc. 7th Int. Symp. Resilient Control Syst.*, Aug. 2014, pp. 1–8.

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

[11] 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.

[12] M. Keshk *et al.*, "Privacy preservation intrusion detection technique for scada systems," in *Proc. Mil. Commun. Inf. Syst. Conf.*, Nov. 2017, pp. 1–6.

[13] 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. 32 910–32 924, 2018.

[14] 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.–Mar. 2021.

[15] D. Upadhyay *et al.*, "Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 1, pp. 1104–1116, Mar. 2021.

[16] U. Adhikari *et al.*, "Industrial control system (ics) cyber attack datasets," datasets used in the experimentation. [Online]. Available: https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets

[17] S. Ghosh and S. Sampalli, "A survey of security in scada networks: Current issues and future challenges," *IEEE Access*, vol. 7, pp. 135812–135831, 2019.

[18] E. D. Knapp and R. Samani, *Applied Cyber Security and the Smart Grid: Implementing Security Controls into the Modern Power Infrastructure*, 1st ed. Syngress Publishing, 2013.

[19] S.-J. Kim *et al.*, "Network anomaly detection for m-connected scada networks," in *Proc. 8th Int. Conf. Broadband Wireless Comput., Commun. Appl.*, 2013, pp. 351–354.

[20] A.-S. K. Pathan, *The State of the Art in Intrusion Prevention and Detection*. Boston, MA, USA: Auerbach Publications, 2014.

[21] Y. Yang *et al.*, "Rule-based intrusion detection system for scada networks," in *Proc. 2nd IET Renewable Power Gener. Conf.*, Sep. 2013, pp. 1–4.

[22] C.-C. Sun, A. Hahn, and C.-C. Liu, "Cyber security of a power grid: State-of-the-art," *Int. J. Elect. Power Energy Syst.*, vol. 99, pp. 45–56, 2018.

[23] S. Tan, "Neighbor-weighted k-nearest neighbor for unbalanced text corpus," *Expert Syst. Appl.*, vol. 28, no. 4, pp. 667–671, 2005.

[24] L. Maglaras, "Intrusion detection in scada systems using machine learning techniques," Ph.D. dissertation, Dept. Comput. Informat., Univ. Huddersfield, U.K., 2018.

[25] Z. Xu *et al.*, "Gradient boosted feature selection," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining (KDD '14)*, 2014, pp. 522–531.

[26] R. Singh, M. Kalra, and S. Solanki, "A hybrid approach for intrusion detection based on machine learning," in *Proc. Int. Conf. Intell. Sustain. Syst.*, 2019, pp. 187–192.

[27] L. Rokach, "Ensemble-based classifiers," *Artif. Intell. Rev.*, vol. 33, no. 1, pp. 1–39, 2010.

[28] J. Surowiecki, *The Wisdom of Crowds*. New York, NY, USA: Anchor Books, 2005.

[29] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[30] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. ser. Springer Series in Statistics. Berlin, Germany: Springer, 2009.

[31] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992.

[32] J. Heinermann and O. Kramer, "Machine learning ensembles for wind power prediction," *Renewable Energy*, vol. 89, pp. 671–679, 2016.

[33] O. Kramer, F. Gieseke, and B. Satzger, "Wind energy prediction and monitoring with neural computation," *Neurocomput.*, vol. 109, pp. 84–93, Jun. 2013.

[34] Y. Wang, Y. Shen, and G. Zhang, "Research on intrusion detection model using ensemble learning methods," in *Proc. 7th IEEE Int. Conf. Softw. Eng. Service Sci.*, 2016, pp. 422–425.

[35] N. T. Pham *et al.*, "Improving performance of intrusion detection system using ensemble methods and feature selection," in *Proc. Australas. Comput. Sci. Week Multiconference*, 2018, pp. 1–6.

[36] F. Idrees *et al.*, "Pindroid: A novel android malware detection system using ensemble learning methods," *Comput. Secur.*, vol. 68, pp. 36–46, 2017.

[37] A. H. Mirza, "Computer network intrusion detection using various classifiers and ensemble learning," in *Proc. 26th Signal Process. Commun. Appl. Conf.*, 2018, pp. 1–4.
[38] M. Abirami, U. Yash, and S. Singh, "Building an ensemble learning based algorithm for improving intrusion detection system," in *Artif. Intell. Evol. Comput. Eng. Syst*. Springer, 2020, pp. 635–649.
[39] X. Li *et al.*, "Sustainable ensemble learning driving intrusion detection model," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 4, pp. 1591–1604, Jul./Aug. 2021.
[40] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," *IEEE Access*, vol. 7, pp. 82512–82521, 2019.
[41] N. Moustafa *et al.*, "Dad: A distributed anomaly detection system using ensemble one-class statistical learning in edge networks," *Future Gener. Comput. Syst.*, vol. 118, pp. 240–251, 2021.
[42] N. Littlestone and M. Warmuth, "The weighted majority algorithm," *Inf. Computation*, vol. 108, no. 2, pp. 212–261, 1994. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0890540184710091
[43] A. Kusiak, H. Zheng, and Z. Song, "Short-term prediction of wind farm power: A data mining approach," *IEEE Trans. Energy Convers.*, vol. 24, no. 1, pp. 125–136, Mar. 2009.
[44] S. Hassan, A. Khosravi, and J. Jaafar, "Examining performance of aggregation algorithms for neural network-based electricity demand forecasting," *Int. J. Elect. Power Energy Syst.*, vol. 64, pp. 1098–1105, 2015.
[45] X. Lin, X. Zhang, and X. Xu, "Efficient classification of hot spots and hub protein interfaces by recursive feature elimination and gradient boosting," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 17, no. 5, pp. 1525–1534, Sep.–Oct. 2020.
[46] C. Strobl *et al.*, "Bias in random forest variable importance measures: Illustrations, sources and a solution," *BMC Bioinf.*, vol. 8, no. 1, p. 25, 2007.
[47] K. M. Ting, *Precision and Recall*. Boston, MA, USA: Springer, 2010, pp. 781–781. [Online]. Available: https://doi.org/10.1007/978-0-387-30164-8_652

Darshana Upadhyay received the master's degree in computer science from Nirma University, Ahmedabad, India. She is currently working toward the Ph.D. degree at the Faculty of Computer Science, Dalhousie University, Halifax, NS, Canada. Prior to starting her Ph.D., she worked as a Lecturer at Nirma University, Ahmedabad, India. For her master's thesis, she has completed a novel project in the area of linear feedback shift register design for secure systems. Her primary research includes algorithm conceptualization, hardware design in the field of embedded systems, vulnerability assessments, and intrusion detection techniques for IoT/SCADA based systems. She was the co-recipient of the Indo-Canadian Shastri research grant in the field of wireless security and intrusion detection systems. She has been invited to be one of the Women in International Security - Canada's 2020 Emerging Thought Leaders. She was awarded the 2020–2021 Citizenship Award from the Faculty of Computer Science, Dalhousie University, for being known as a congenial, reliable, mature person who is respected by peers, and for building a community atmosphere within the faculty. She was also awarded the Gold Medal for securing the first position during her master's degree.

Jaume Manero received the Ph.D. degree in artificial intelligence from the Technical University of Catalonia, Barcelona, Spain. He is currently with the Barcelona Supercomputing Center, Barcelona, Spain. His Ph.D. dissertation was deep learning architectures applied to wind time series forecasting. He is also a Visiting Research Scientist with Dr. Srini Sampalli's MYTech Lab (Emerging Wireless Technologies) where he is working on how deep learning can impact in the development of cyber-security applications.

Marzia Zaman received the M.Sc. and Ph.D. degrees in electrical and computer engineering from the Memorial University of Newfoundland, St. John's, NL, Canada, in 1993 and 1996, respectively. In 1990, she joined the Nortel Networks, Ottawa, ON, Canada, where she joined the Software Engineering Analysis Lab 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 Software Designer with Accelight Networks, Excelocity, Sanstream Technology, and Cistel Technology, Ottawa, ON, Canada. 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, on multiple power engineering projects. Her research interests include renewable energy, wireless communication, IoT, cyber security, machine learning, and software engineering.

Srinivas Sampalli (Member, IEEE) received the bachelor of engineering degree from Bangalore University, Bangalore, India and the Ph.D. degree from the Indian Institute of Science, Bangalore, India, and is currently a Professor and National 3M Teaching Fellow in the Faculty of Computer Science, Dalhousie University, Halifax, NS, Canada. He has led numerous industry-driven research projects on Internet of Things, wireless security, vulnerability analysis, intrusion detection and prevention, and applications of emerging wireless technologies in healthcare. He currently oversees and runs the Emerging Wireless Technologies Lab and has supervised over 150 graduate students in his career. His primary joy is in inspiring and motivating students with his enthusiastic teaching. He is the recipient of the Dalhousie Faculty of Science Teaching Excellence Award, the Dalhousie Alumni Association Teaching Award, the Association of Atlantic Universities' Distinguished Teacher Award, a teaching award instituted in his name by the students within his Faculty, and the 3M National Teaching Fellowship, Canada's most prestigious teaching acknowledgement. Since September 2016, he holds the honorary position of the Vice President (Canada), of the International Federation of National Teaching Fellows, a Consortium of National Teaching Award winners from around the world.