Intrusion Detection in SCADA-based Power Grids: Feature Selection using Gradient Boosting Scoring Model with Decision Tree Classifiers

Darshana Upadhyay
Faculty of Computer Science
Dalhousie University,Halifax, Canada
City, Country
darshana@dal.ca

Jaume Manero
Faculty of Computer Science
Dalhousie University
Halifax,CANADA
Jaume.Manero@dal.ca

Marzia Zaman
Research & Developmnt Dpt
Cistel Technology
Ottawa, Canada
Marzia@cistel.com

Srinivas Sampalli
Faculty of Computer Science
Dalhousie University, Halifax, Canada
City, Country
srini@cs.dal.ca

***Abstract*—Smart grids rely on SCADA (Supervisory Control and Data Acquisition) systems to monitor and control complex electrical networks in order to provide reliable energy to homes and industries. However, the increased inter-connectivity and remote accessibility of SCADA systems expose them to cyber attacks. As a consequence, developing effective security mechanisms is a priority in order to protect the network from internal and external attacks. We propose an integrated framework for an Intrusion Detection System (IDS) for smart grids which combines feature engineering-based preprocessing with machine learning classifiers. Whilst most of the machine learning techniques fine-tune the hyper-parameters to improve the detection rate, our approach focuses on selecting the most promising features of the dataset using Gradient Boosting Feature Selection (GBFS) before applying the classification algorithm, a combination which improves not only the detection rate but also the execution speed. GBFS uses the Weighted Feature Importance (WFI) extraction technique to reduce the complexity of classifiers. We implement and evaluate various decision-tree based machine learning techniques after obtaining the most promising features of the power grid dataset through a GBFS module, and show that this approach optimizes the False Positive Rate (FPR) and the execution time.**

***Index Terms*—SCADA Systems, power grids, random forest, gradient boosting, feature selection, cyber security, network intrusions**

I. Introduction

POWER grids are the basic infrastructure that support our economies and daily lives by providing and sustaining a continuous supply of electricity. They play a fundamental role in connecting our industries and homes with locations far away from where the electricity is generated, while assuring the quality of the electricity supply at the point of consumption.

These systems are complex and distributed in nature and comprise several components such as power lines, transformers, sensors, phasor measurement units (PMUs) and substations connected to supervisory control and data acquisition (SCADA) systems for real time monitoring, management and control. Figure 1 illustrates the block diagram of a SCADA architecture for a power grid, showing SCADA components such as SCADA Master, HMI, PLCs, RTUs , and various power grid components such as IEDs, substation switch and control room components.

Generally, the sensors and PMUs at power stations monitor different attributes of electrical signals continuously and transmit that to the field control devices such as PLC, RTU, or IED. Communication between the field control devices and the SCADA master takes place via communication links and switches. The SCADA master is located at the control center.

The field control devices supply digital status information to the SCADA Master to determine acceptable parameter ranges. This information will then be transmitted back to the field device(s) where action may be taken to optimize the performance of the system. Moreover, the status information is stored in a data historian and displays it on an HMI (Human Machine Interface), which provides centralized monitoring and system control.

Originally, power grids were designed to generate and distribute the electricity in an efficient and timely manner, rather than focusing on security aspects of the critical infrastructure of the system. However, the increase of inter connectivity and remote accessibility places power grids under the risk of internal and external attacks.

Real-time cyber attacks can disrupt entire power grids. For example, in 2003 the Davis-Besse nuclear power plant near Oak Harbor, Ohio was infected by a Slammer worm that traveled from a consultant's network to the process control network and generated unwanted traffic [1]. As a result, the plant personnel could not access the safety parameter display system for around five hours which showed sensitive data about the reactor core, temperature, and radiation sensors of

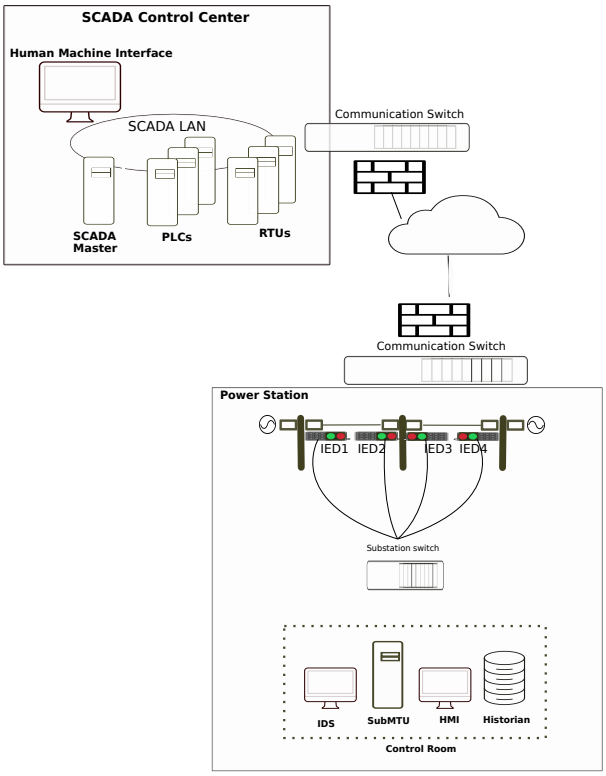


Figure 1: SCADA System Architecture for Power Grids
Legend: PLCs: Programmable Logic Controllers, RTUs: Remote Terminal Units, HMI: Human Machine Interface, IEDs: Intelligent Electronic Devices

the power plant. In 2006 the Browns Ferry nuclear plant in Athens, Alabama was shut down after the failure of critical reactor components and controllers due to a cyber attack on their internal network [2]. In 2008, the second unit of the Hatch nuclear power plant in Baxley, Georgia experienced an automatic shutdown due to routine software update to a single computer on the plant floor. The update was performed to synchronize data between the plant and business networks [2]. Another incident in an Iranian nuclear plant was reported in 2011 where the plant process was interrupted due to the Stuxnet worm. This attack was initiated by connecting an infected USB drive to the Programmable Logic Controller (PLC) at the plant floor [3]. The Ukraine power plant cyber attack was reported in 2015 [4]. This was the first known successful attack on power grids where attackers were able to disrupt electricity supply to the end users. Thus, power grid attacks are one of the most critical issues in industrial control systems and it is important to protect them by applying adequate safety measures [5].

General safeguards include defense-in-depth architecture which separates the control and corporate network traffic, strong access control and authentication mechanisms, restricted perimeters using DMZ (demilitarized zone), vulnerability assessment and risk management systems [6]. However, these safeguards are difficult to deploy and maintain owing to legacy-inherited security loopholes and restrictions [7]. Therefore, these relevant preemptive measures are not sufficient to protect the power grids from cyber attacks. Additional protection layer is also required which detects and prevents the system from malicious events and threats.

Generally, packet filtering and identification of threats are key to securing these systems. However, traditional firewalls do not always fulfill all the security requirements of critical infrastructures. For example, in 2019, the western US power grid infrastructure was hacked. The intruders created periodic blind spots for grid operators for about 10 hours, by identifying a vulnerability in the firewall configuration [8]. Therefore, the design and development of sophisticated and accurate intrusion detection and prevention systems are one of the primary objectives to secure power grids.

Researchers and security experts have proposed various intrusion detection and prevention approaches to ensure secure and safe operations of power grids. A signature-based approach is used for pattern matching to determine frequent signatures of malicious packets [9]. In this approach the signature of every incoming packet is compared with all the stored signatures to identify threats. This approach is valid for known intrusions but is unable to identify zero-day attacks [9].

More recently, data mining, clustering and statistical signal processing approaches have been used for anomaly detection. These techniques are effective compared to pattern-matching, but usually generate a high level of false-positive alarms [10]. Therefore, there is a need for better techniques that detect intrusions from real incoming traffic. Machine Learning and Deep Learning have stronger pattern recognition capabilities than standard approaches. These techniques train and test the model according to real network traffic to detect anomalies with better precision and generate a smaller number of false-positive alerts. Some of the most prominent machine learning techniques include decision trees, Bayesian, genetic algorithms, neural-networks and support vector machines [11].

Decision tree algorithms, which make decisions using bias and variance analysis mechanisms are one of the powerful supervisory machine learning techniques. Furthermore, ensemble methods use the principle of combining weak learners to obtain a stronger 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. Moreover, Gradient Descent is used to optimize the overall tree selection. This combined approach provides a powerful method for identification and pattern recognition capabilities for structured data [12].

Our proposed approach uses the Gradient Boosting algorithm as the base classifier to detect malicious activities in power grids. To solve the classification and regression problems, the ensemble Gradient Boosting algorithm has proven to be more efficient than traditional boosting approaches [13].

The ensemble Gradient Boosting algorithm is an ensemble learning method based on a combination of additive models (weak learners), which can gradually learn from the previous misclassifications to create a stronger learning model [14]. This algorithm has been complemented with a feature selection process that increases the overall performance by extracting the most relevant features from the input data.

The proposed technique has been developed using various library functions of the open source library scikit-learn [15]. The library offers various classification, regression, and clustering algorithms. Table I summarizes the general scientific meanings of the software implementation terms used in this paper.

Table I: General scientific meanings of software implementation terms

Term	Description
scikit- learn	This is a Python module integrated with a wide range of machine learning techniques for both supervised and unsupervised learning. We have used various functions of scikit-learn library for the implementation and comparative analysis of the proposed methodology.
Num_trees	This term indicates the number of trees that we want to build for the average prediction. For the proposed feature selection technique, we have tuned this parameter by creating 100, 500, 700, and 1000 trees at each iteration.
Model based feature selection	This refers to the meta-transformer technique, which uses the WFI scoring model to remove insignificant features according to the threshold value. In this paper, we have used Grading Boosting as a base model, and the threshold value is set to 0.5 to remove the unimportant features.

The major contributions of this paper are as follows.

1) We use the gradient boosting weighted feature importance scoring model and tune the Num_trees parameters to identify the top important features. To make it more efficient, we merge these two concepts to select the most promising and common features from the existing datasets that reduce the overhead and increase the execution speed for SCADA based power grids.
2) We derive 15 most promising features from the binary class and apply the same features to the rest of the three categories, namely, three class, seven class, and multi class, to evaluate the performance of the feature selection module.
3) We evaluate eight different tree-based algorithms to validate the effectiveness of the selected features for the classification of various power system attacks.
4) We perform a comparative analysis of eight tree-based classifiers and identify the top three tree-based classifiers according to multiple performance metrics.
5) We compare the accuracy of proposed methodology with published state-of-the-art techniques.

The rest of this paper is organized as follows. Section II describes related research in the area of power grid security by considering various attacks and protection schemes. The proposed intrusion detection system framework based on Gradient Boosting Feature Selection is introduced in Section III. Section IV covers algorithm conceptualization and mathematical proof of our approach. Section V describes the proposed mechanism of feature selection by combining regularization strategies with Weighted Feature Importance metrics. Section VI presents the complete experimental setup, evaluations, result-analysis and comparative studies of various tree-based machine learning techniques performed on power grid datasets. Conclusion and future work are provided in Section VII.

II. BACKGROUND AND RELATED WORK

Many researchers have proposed different types of intrusion detection systems (IDSs) according to the need of securing various components in power grids. For example, one approach is specifically focused on security of the RTU and the PLC, as these devices are easy targets for cyber attacks [16]. A real-time attack with malware running on a PLC was demonstrated by black hat researchers in 2016 [17].

Malicious cyber-attacks have costly consequences in power grids, and as a result the grid operators are increasingly investing in IDSs. IDSs are typically based on the principle that attacks show different behavior and patterns from the normal traffic [18]. In this sense the classification problem can be reduced to a pattern recognition activity. To identify malicious behavior, identifying a pattern that differs from the normal flow is required. The traditional approach is to develop a signature of the attack and recognize this signature. This method requires extensive manual work as the signature is manually added to the database when the attack is identified and its signature extracted. A more sophisticated approach is to use machine learning to perform the pattern recognition process [11].

Feature selection is also known as dimensionality reduction, which is used to improve the accuracy of estimators and boost the performance of the high-dimensional datasets. The feature selection techniques are mainly categorized into four types, namely, Variance Threshold (VT), Univariate Feature Selection (UFS), Recursive Feature Elimination (RFE) and Model based feature selection. VT is a simple baseline approach that removes all the variance which does not meet the threshold, whereas UFS follows the method of a statistical test to identify the best features [19]. In the UFS approach, the features are selected by either comparing false positive rates or obtaining scores or percentile of the given features [20]. Moreover, the configurable strategy of UFS allows a combination of two approaches, namely, univariate selection and hyper-parameter search estimator.

On the other hand, RFE selects the features recursively by comparing the outcome of a larger set with the smaller set while training the dataset [21]. This technique is more efficient in terms of estimators' accuracy scores but computationally costlier than VT and UFS. Model based feature

selection method is a meta-transformer that uses the WFI scoring model to remove unimportant features according to the threshold value [22]. This is comparatively faster than other techniques as feature importance score is obtained during tree construction. Moreover, this method can easily be merged with other estimators, such as tuning the parameters.

To identify top features, we have used a gradient boosting based WFI scoring model to discard the irrelevant features along with Num_trees to tune the parameters. This approach improves not only the accuracy of the tree-based classifiers but also the execution speed.

Several machine learning approaches have been tested to filter malicious packets, for instance, K-nearest neighbours (k-NN) is quite effective, since its main characteristic, is being a 'lazy learner' - it does not contain the trained model but builds it real-time by learning from the nearest neighbours - is very well aligned to this task. However it has high performance requirements and may have fitting issues for imbalanced small datasets [23]. Other tested approaches such as Support Vector Machines (SVM), which maps the inputs into another dimensional space, offer good results, but are costly to train. Neural network approaches have also shown strong representational capabilities, but have not yet been widely applied in commercial applications [11].

In the classification field, and for structured data inputs, the gradient boosting family of algorithms shows improved representation capabilities [24]. This approach combines boosting with decision trees techniques. Specifically they combine random tree refinements with boosting techniques' optimization. Variants like Gradient Tree Based Boosting (GTBM) or the recently developed XGBoosting (Extreme Gradient Boosting) are becoming tools of choice in many applications [24]. However their effectiveness has not been widely studied on various IDS applications, which is the main motivation for this work.

Furthermore, power grid SCADA systems rely on real time request response mechanisms to operate the sub-station components accurately by consuming minimal CPU and battery resources. For such time-critical systems, the deployed intrusion detection system should act as quickly to capture malicious activities using minimal resources in a given time period for larger-scale deployments. Our proposed model leverages all the competencies for such systems. The model offers a combination of efficiency with precision, as it reaches high accuracy levels while using a limited amount of resources. This combination makes this model a good fit for mission critical applications or for large sets of disseminated SCADA devices, that have limited computing availability for filtering mechanisms, and both these properties fit very well with the power system scenario.

III. Framework for a GBFS based Intrusion Detection System

This section presents the proposed framework for an intrusion detection system that distinguishes normal and malicious events by analyzing SCADA traffic on power grids. The proposed framework operates in three phases, namely, preprocessing the data, feature selection, and anomaly detection using a classification approach. The elements for each phase are illustrated in Figure 2.

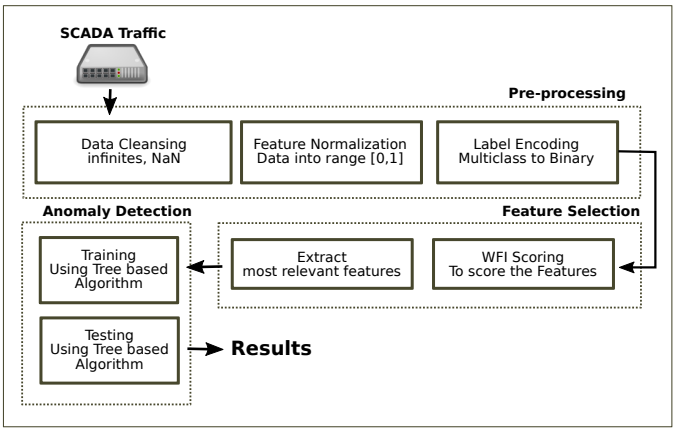


Figure 2: Framework for a GBFS Based Intrusion Detection System

During the data preprocessing phase, data cleansing, feature mapping and feature normalization are applied to the raw dataset to obtain filtered data. Then the Gradient Boosting Feature Selection approach is applied on filtered data to select the most promising features from the entire dataset dynamically. Since power grids use a complex mix of SCADA systems to control field-site components, network monitoring devices such as SNORT and Syslog are used to capture the different types of features [25].

Usually, real-time data obtained from sensors or real-time systems always presents some consistency issues, the signal is lost, or the measuring devices get off the scale readings at some point. For this reason, we need to do a data cleansing operation to remove incorrect data. We remove infinities and NaN values, looking for empty sequence points that will be avoided by the algorithms.

Furthermore, in order to extract the relevant features, we apply a Gradient Boosting Feature Selection which uses Weighted Importance Feature extraction method to select the most promising features. This approach helps to improve the computational speed and also assists in providing a precise outcome for anomaly detection. Moreover, reduction in features helps in consuming less memory while training and testing the dataset during classification to classify normal and attack events.

IV. Gradient boosting and XG Boosting Theory

In this work, we have used the combination of two main concepts, namely, the gradient boosting WFI scoring model and Num_trees for feature selection, and XGBoost as one of the classification methods.

One of the most efficient techniques of the tree-based ensemble method is called boosting, which stores the labels

and weights of the leaf nodes that make the prediction interpretations easy to handle. Gradient boosting [26] is a practical approach proposed by Chen et al [24] and is considered as one of the algorithms of choice in machine learning. We can obtain a strong learner by combining weak learners during the gradient boosting process. In this technique, the classification is dependent on the residuals of the previous iteration where the impact of each feature is evaluated sequentially until a target accuracy is obtained. The residuals are calculated by a Loss function $\mathcal{L}(\phi)$ that is optimized using gradient descent. The final result $\phi(X)$ is obtained by the addition of the results of the K sequential classifier functions f_k as follows:

$$\hat{Y} = \Phi(X) = \sum_{k=1}^{K} f_k(X) \quad f_k \in F \tag{1}$$

where f_k is a decision tree, and K is the total number of iterations in the boosting algorithm.

XGBoosting has two enhancements, an improvement over Gradient Descent and a more sophisticated regularization strategy. The regularization factor to the cost function controls the optimization process and manages the overfitting factor. In this, the function to optimize in Step t is called the regularization term $\Omega(f_t)$ and we use it in the following equation to calculate a Loss function $\mathcal{L}(\phi)_t$ at step t.

$$\mathcal{L}(\phi)_t = \sum l(f_{t-1} + f_t) + \Omega(f_t) \tag{2}$$

Without the regularization factor, the tree will split until it learns all the features of the training set, which may result in overfitting. By using a regularization function the training stops when the function identifies that the model is good enough based on the learning score, which avoids the chance of overfitting.

During optimization, the regularization term is improved by approximation using a short Taylor series decomposition. For complete details of XGBoost, we refer the reader to the original article written by Chen et al [24].

A. Using Weighted Feature Importance (WFI) for Feature Selection

Gradient boosting uses a powerful metric, called $feature/importance$, to retrieve the scores of each attribute according to importance after the boosted tree is constructed. This scoring model provides the importance of each attribute in terms of making key decision while constructing decision trees. Generally, feature importance provides a score that defines the significant role of each attribute. This importance is computed explicitly by comparing and ranking all the features amongst one another in the dataset. The importance of a single decision tree is calculated by the amount of each attribute split point, weighted by the number of observations from that node. This split point is used to improve the performance and efficiency of the algorithm.

In particular, purity (Gini Index) is used to select the split points or to identify a more specific error function. The feature importance of each tree is averaged across all the decision trees within the model. The Model based feature selection class is used to transform a dataset into subsets by using the most promising features. The focal point of this approach is to embed the preprocessing with this model using WFI to reduce the training time by removing irrelevant features from the given dataset. Once the most promising ones are derived through the GBFS technique, we can effectively use them for training and testing the model.

V. A novel Weighted Featured Selection algorithm for intrusion detection

A. Power System – Testbed Description

This section describes the overall approach with regard to multilevel multiple attack vector classification of power system disturbances. To evaluate the performance of the GBFS based proposed algorithm, three publicly available datasets are used [27]. These datasets were created at Oak Ridge National Laboratories (ORNL) using the power system testbed.

The power system testbed configuration has been implemented using power generators- G1,G2 and IEDs - R1 through R4, to control the breakers BR1 through BR4, on or off, respectively. To fulfill the simulation requirements, the three-bus two-line transmission system is created [28]. Each one of the four IEDs uses a distance protection scheme to trip the respective breaker in case of fault detection, whether the nature of the fault is valid, or faked since they do not have smart logic to detect the difference between original and fake faults. Furthermore, operators can manually trip the breakers by issuing commands in case of maintenance on the lines or other system components [25].

B. Dataset

The datasets include measurement related to normal, disturbance, control and cyber attack behaviours with regards to electrical transmission system in the power grid [29]. There are three publicly available datasets and two of them are derived using the third main dataset consisting of fifteen sets with 37 power event scenarios in each dataset. The datasets are randomly sampled and categorised into three major classes; Binary, Three-class and Multiclass. Furthermore, we have derived a fourth dataset named Seven-class of fifteen sets from the Multiclass dataset, consisting of seven power event scenarios in each.

The experiments were carried out using 4 different categories of the datasets where the Binary dataset has two output labels, namely, normal and attack, The Three-class dataset has three output labels - one additional label to binary dataset is no event. The Seven-class dataset has seven output labels as follows: 1 natural SLG (Single Line Ground) fault event owing to short-circuit in a power line, 1 data injection attack, 2 remote tripping command injection attacks and, 3 relay setting change attacks. The 37 scenarios of Multiclass dataset are divided mainly in three categories - Natural events, 1 No event and 28 Attack events. 8 Natural events categorized in 6 SLG faults and 2 Line maintenance events. Furthermore, no event indicates normal operation load changes and 28 attack events

are mainly divided into 3 major attack events termed as Data Injection, Remote Tripping Command Injection and Attack on Relay Setting. These attacks are further subcategorized in 6 data-injection SLG fault replay attacks, 4 command injection attacks against single IED (relay), 2 command injection attacks against two IEDs, 10 relay setting change attacks on a single IED, 4 relay setting change attacks on two IEDs, and 2 relay disable and line maintenance attacks [30]. Moreover, these authentic datasets are used in various experiments related to power system cyber-attacks classification [27]. All the attacks scenarios are simulated by assuming that the intruder is an internal entity, which is capable enough to launch various attacks by issuing malicious commands from the substation switch [25].

Each power grid dataset consists of 128 features. To derive these features, 4 phasor measurement units (PMUs) are used to measure the electrical signals on an electrical power grid using common time source to maintain time synchronization. Each PMU measures 29 features, hence in total 116 PMU measurement carried out using 4 PMUs. These features are referred as R# - signal_Reference which indicates the index of PMU and type of measurement. For example, R1-PA1:VH represents the Phase A voltage phase angle measured by PMU R1 [27]. Also, 16 more columns are additionally inserted by control panel logs, snort alerts and relay logs where relay and PMU are integrated together [30]. The last column represents the marker to label different events. The description of all the features is shown in Table II. Also, each set of 15 sets consist average 294 "no event" instances, 1221 natural events instances and 3711 attack vectors across the classification schemes [25].

Table II: Description of features

Feature	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

C. Regularization Strategies

Generally, boosting algorithms play a vital role in controlling the bias-variance trade-off. The objective of the gradient boosting algorithm is to generate an optimal combination of the trees while training the model using the concept of binomial deviance theorem. In addition to minimizing the loss function to the smallest possible degree, it is necessary to tune the hyper-parameters carefully, since complex trees overfit and simple trees can move the model to under-fitting. The majority of the tuning parameters are divided into two categories, one is specifically meant for construction and efficiency of each individual tree, and the other type of boosting parameters are used to boost the operation in the model. Owing to this fact, we have tuned the hyper-parameters by extensive grid search, taking learning rate, sub-samples and Num_trees into consideration.

We have analyzed the effect of different regularization strategies on various datasets by implementing a grid search. Figure 3 illustrates the effect of boosting parameters of one of the 15 binary datasets. According to the results depicted in the graph, regularization via shrinkage (learning rate = 0.1) improves the performance significantly, as compared to without shrinkage (learning rate & subsample = 1.0) and in the case of stochastic gradient boosting (combination with learning rate and subsample < 1.0).

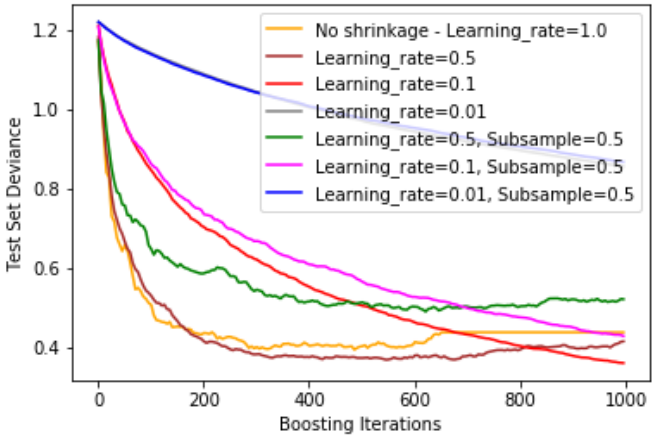


Figure 3: Different Regularization strategies applied on a binary classification. A hyper parameter optimization (learning rate = 0.1) improves the result significantly, with small learning rates more trees are required for convergence

D. Feature Selection

Generally, when we have a big model with hundreds or thousands of features, the feature selection approach is used to choose the most promising features and to remove irrelevant features while retraining the model. Also, by analyzing the importance of each feature manually, we can get an idea of what the model is doing, and the model is working well. Here, we derive the importance of each feature by applying WFI scoring method on Gradient Boosting trained model. Furthermore, all the features are depicted as a percentage rating of how often the feature is used in determining the output label. To make the list of features easier to read, we have sorted them from most important to least important as shown in Figure 4.

The feature importance scores reflect information gain by each feature during the construction of a decision tree. During experiments, we observe 50% of the 128 features are not contributing to making any decision. The WFI score of such

Table III

Gradient Boosting Feature Selection (Best 15 Features of 15 Datasets for all the four categories - Binary, Three classes, Seven classes and Multi-class)

features	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13	f14	f15
D1	R2-PA3:VH	R1-PM10:I	R1-PA1:VH	R2-PM1:V	R2-PA10:IH	R2-PA5:IH	R2-PM10:I	R3-PA5:IH	R1-PM5:I	R3-PA1:VH	R2-PM5:I	R1-PA5:IH	R4-PA1:VH	R3-PA7:VH	R1-PA7:VH
D2	R2-PM3:V	R4-PA2:VH	R1-PA2:VH	R2-PM1:V	R4-PA1:VH	R1-PA5:IH	R4-PM2:V	R1-PA1:VH	R3-PM7:V	R2-PA3:VH	R2-PA7:VH	R4-PM1:V	R3-PA5:IH	R3-PM5:I	R1-PA7:VH
D3	R3-PA4:IH	R2-PM10:I	R3-PA2:VH	R2-PA2:VH	R2-PM5:I	R4-PA1:VH	R2-PA4:IH	R3-PM2:V	R2-PA5:IH	R1-PM5:I	R3-PA3:VH	R2-PA3:VH	R3-PM5:I	R1-PA7:VH	R4-PM5:I
D4	R2-PA7:VH	R4-PM5:I	R4-PA2:VH	R4-PA3:VH	R4-PA7:VH	R1-PA5:IH	R4-PM2:V	R2-PA2:VH	R2-PA5:IH	R4-PA1:VH	R1-PM5:I	R4-PA5:IH	R1-PA3:VH	R1-PA2:VH	R2-PM5:I
D5	R3-PA7:VH	R1-PA5:IH	R4-PM2:V	R4-PA4:IH	R4-PA5:IH	R4-PM5:I	R3-PM6:I	R3-PA10:IH	R2-PA5:IH	R3-PA3:VH	R1-PA3:VH	R4-PM4:I	R4-PA1:VH	R2-PA2:VH	R4-PA7:VH
D6	R4-PA1:VH	R3-PM2:V	R3-PA2:VH	R4-PM3:V	R1-PA2:VH	R4-PA7:VH	R2-PA10:IH	R4-PA2:VH	R2-PA5:IH	R1-PM10:I	R1-PA7:VH	R4-PM2:V	R3-PA5:IH	R4-PM5:I	R1-PM5:I
D7	R4-PA6:IH	R1-PA7:VH	R1-PM5:I	R1-PA1:VH	R2-PM7:V	R1-PA6:IH	R4-PA7:VH	R3-PA5:IH	R3-PA6:IH	R4-PA1:VH	R4-PA3:VH	R3-PM2:V	R4-PM7:V	R2-PA3:VH	R3-PA3:VH
D8	R4-PA7:VH	R1-PM2:V	R1-PA2:VH	R2-PA3:VH	R1-PA5:IH	R2-PA1:VH	R1-PM5:I	R1-PA3:VH	R3-PA5:IH	R3-PA6:IH	R3-PA2:VH	R4-PA3:VH	R4-PM2:V	R4-PA1:VH	R4-PA5:IH
D9	R2-PA2:VH	R4-PM7:V	R2-PM5:I	R4-PA1:VH	R3-PA2:VH	R1-PA3:VH	R4-PA7:VH	R3-PA5:IH	R1-PM2:V	R1-PA2:VH	R4-PA5:IH	R2-PA3:VH	R3-PA3:VH	R4-PA2:VH	R4-PM2:V
D10	R3-PA4:IH	R1-PA1:VH	R1-PA7:VH	R4-PA5:IH	R4-PA7:VH	R2-PM1:V	R1-PA5:IH	R4-PA1:VH	R4-PM5:I	R4-PM7:V	R3-PM2:V	R2-PA5:IH	R4-PA2:VH	R4-PM2:V	R3-PA5:IH
D11	R2-PA4:IH	R3-PA5:IH	R4-PM1:V	R1-PM5:I	R2-PM5:I	R2-PA1:VH	R4-PM5:I	R2-PM7:V	R1-PA2:VH	R2-PA6:IH	R2-PA5:IH	R4-PA2:VH	R2-PM1:V	R4-PM7:V	R4-PM2:V
D12	R4-PA3:VH	R4-PA7:VH	R2-PA3:VH	R1-PA2:VH	R2-PM5:I	R1-PM5:I	R3-PM2:V	R2-PA5:IH	R3-PA5:IH	R4-PA1:VH	R4-PA2:VH	R4-PM5:I	R1-PA:ZH	R1-PA3:VH	R3-PM5:I
D13	R3-PA3:VH	R3-PA2:VH	R2-PA3:VH	R3-PA5:IH	R4-PA6:IH	R1-PA1:VH	R4-PA7:VH	R4-PA2:VH	R4-PM2:V	R3-PM2:V	R2-PA5:IH	R1-PA7:VH	R1-PA3:VH	R4-PA3:VH	R4-PA1:VH
D14	R3-PM10:I	R1-PA1:VH	R2-PA5:IH	R4-PM7:V	R1-PM2:V	R4-PA5:IH	R1-PA2:VH	R2-PA6:IH	R3-PA3:VH	R3-PA6:IH	R1-PA7:VH	R4-PA3:VH	R1-PA6:IH	R1-PA3:VH	R4-PM2:V
D15	R4-PA1:VH	R4-PM7:V	R2-PM5:I	R1-PA5:IH	R3-PM3:V	R1-PA7:VH	R3-PA5:IH	R4-PM5:I	R2-PA3:VH	R2-PA5:IH	R4-PA7:VH	R1-PA1:VH	R3-PA3:VH	R3-PM2:V	R4-PM2:V

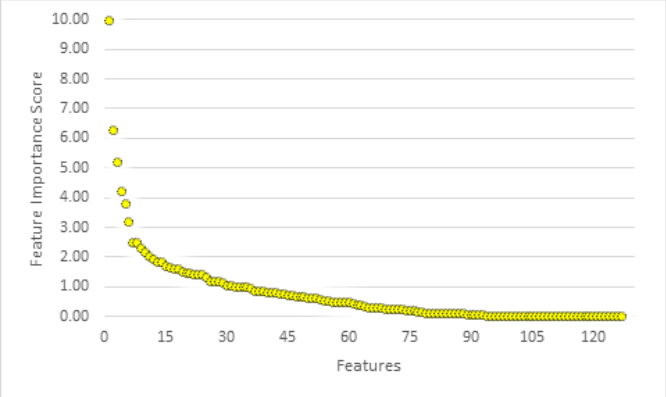


Figure 4: WFI scoring model to rank the features

features is zero. While, out of the remaining 50% of features, 15 features provide a significant contribution in making decisions during the construction of decision-tree. The WFI score of those features has high values in the range of 1 to 10. The rest of the 45 features having feature importance scores between 0 and 1. These 45 additional features contribute comparatively less and have a large drop in feature importance score. Altogether the entire dataset is divided into three levels of information gain groupings, namely, most promising, slightly contributing, and irrelevant features.

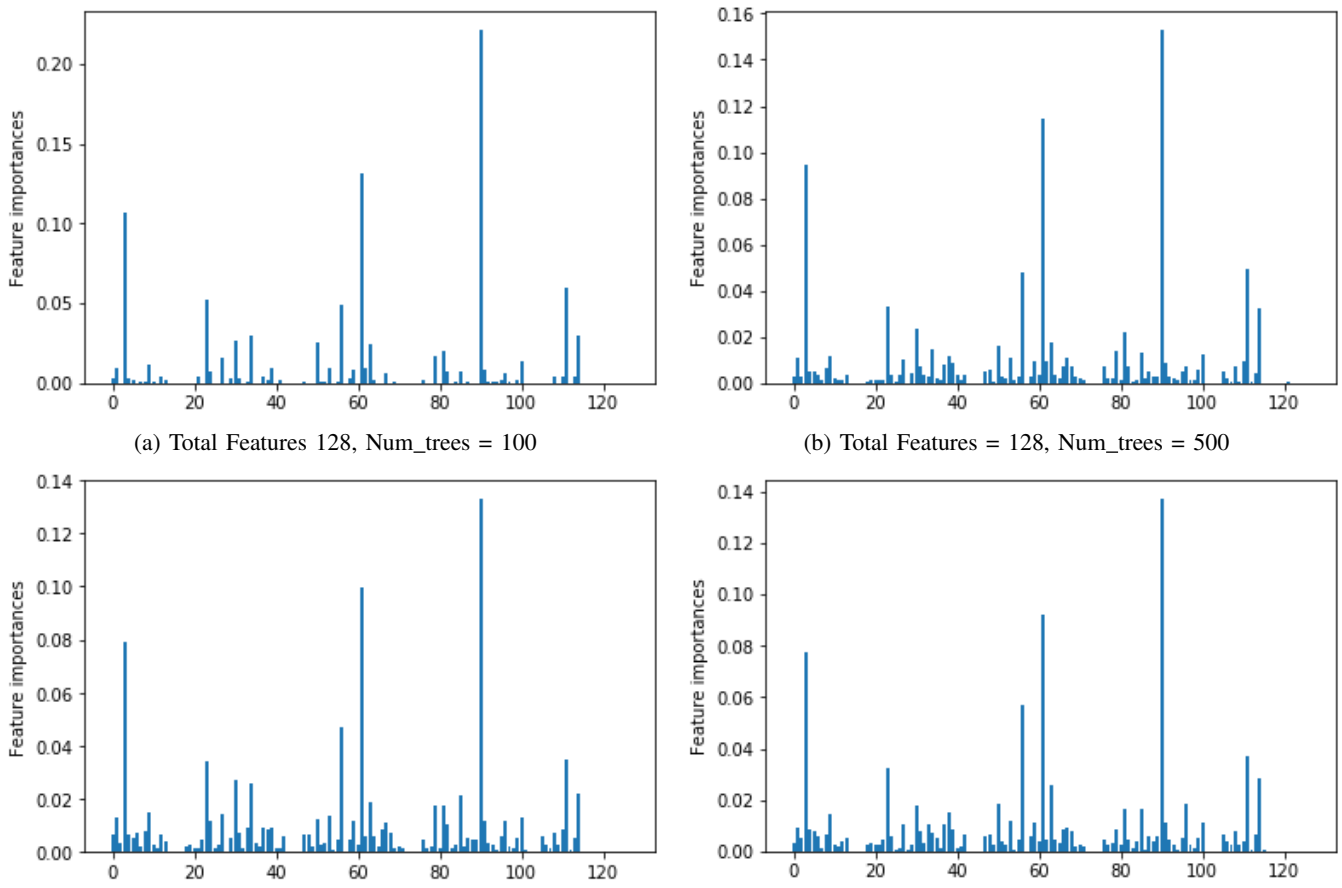
According to [31], feature extraction creates a subset of the given features which not only reduces the noise but also improves the classifiers' performance. Therefore, we have tested 15 datasets of four different categories (binary, three-class, seven-class & Multi-class) of power grid system created by the Oak Ridge National Laboratories using the most promising features [27]. To identify these best features, we use the WFI scoring model along with concept of Num_trees.

Furthermore, to increase the execution speed, we perform feature extraction on binary datasets. We repeat the entire process by taking the various parameter value of Num_trees to collect various observations. From that we have identified best features by taking common important features from the estimations as shown in the Algorithm 1. Here, Num_trees refers to the number of estimators whereas n refers to the total number of features. We have used four estimators, namely, 100, 500, 700, 1000 and initially dataset consist of n = 128 features.

Algorithm 1: Weighted Feature importance based on a Gradient Boosting Feature selection model

Input: Training power-grid dataset PD
Output: Selected feature subset Selected PD
Initialize: Current power-grid dataset
Current-PD=$\{1, 2, \cdots, n\}$;
begin
  i $\leftarrow$ 0
  Num_trees $\leftarrow$ $\{100, 500, 700, 1000\}$
  Num_trees $\leftarrow$ Num_trees (i)
  while *Features(Num_trees > 0)* **do**
    (1) Create GB model on value Num_trees
    (2) Evaluate Ranking with WFI scoring
    (3) Remove features lower importance
    (4) Store the features in Scored-PD
    (5) Num_trees $\leftarrow$ Num_trees (i+1)
  end
  (6) Compare features of Scored-PD from all Num_trees
  (7) Take common features of Scored-PD
    Selected-PD $\leftarrow$ Scored-PD
end

Figure 5 represents the relative importance of each attribute on the binary dataset by considering four estimators. The high vertical bars represent the most promising and common features in all four estimators. In this experiment, all estimators use the top 15 features for each ensemble. In Table III we observe the most promising features across all 15 datasets. Also, to validate the strength of the selected features, the same 15 ones are applied to all four categories (Binary, three classes, seven classes and Multi-class) of intrusion classification. It can be observed that each dataset has a different set of stronger features, a conclusion that points to independent feature selection process for each dataset type. The most important features which contribute in determining the intrusions are Voltage



(a) Total Features 128, Num_trees = 100

(b) Total Features = 128, Num_trees = 500

(c) Total Features = 128, Num_trees = 700

(d) Total Features = 128, Num_trees = 1000

Figure 5: Represents the relative importance of each attribute of the dataset with 5000 records;computed by considering four estimators Num_trees = 100,500,700,1000

Phase Angles, Voltage Magnitude, Current Phase Angles and Current Magnitudes according to the attack location on PMUs.

VI. Experiments

A. Evaluation parameters

The choice of the evaluation parameters always depends on the nature of the dataset, whether it is a multi-class or just binary. Typically, datasets are imbalanced in nature, a property defined by having classes of different sizes. Hence to evaluate the efficiency of the proposed GBFS based framework, our approach does not only relies on the accuracy of the classifier but also incorporates other assessment parameters like Detection Rate (True Positive Rate also called Recall & True Negative Rate), Precision, F1 Score and Miss Rate (False Negative Rate).

The assessment metrics, namely, accuracy, recall, precision and false negative rate depend on the following four parameters,namely, True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) [32]. TP refers to the number of actual attacks which are classified as attacks, TN refers to the number of normal events classified as normal events, FP refers to the number of normal events misclassified as attacks and FN refers to the number of attacks misclassified as normal events. The evaluation metrics are defined as follows, described from the basic four definitions.

- Accuracy is the percentage of all normal and attack vectors that are correctly classified:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

- Detection Rate (True positive Rate (TPR) and True Negative rate (TNR)) refers to the percentage of total relevant results correctly classified by the classifier

$$TPR = \frac{TP}{TP + FN} \text{ (attack vector)} \tag{4}$$

$$TNR = \frac{TN}{TN + FP} \text{ (normal event)} \tag{5}$$

- Precision or Positive Predictive Value (PPV) refers to the percentage of the results which are relevant.

$$PPV = \frac{TP}{TP + FP} \text{ (attack event)} \tag{6}$$

- F1 Score is simply the harmonic mean of precision and

recall evaluating the outcome in balanced mode

$$F1\_score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (7)$$

- Miss Rate (FNR/FPR) is derived by subtracting the value of TPR from 1.

$$FPR = 1 - TNR \text{ (attack)} \quad (8)$$

$$FNR = 1 - TPR \text{ (normal)} \quad (9)$$

B. Experimental results

Our target is to develop a model in such a way that it can be easily deployed in a real-time power grid. For that, the model should be fast and smart in identifying malicious events that occur in the network. Therefore, we target the most relevant features to classify normal and attack vectors. To compute the most promising features, we have used a WFI scoring model of Gradient Boosting feature selection. We have applied the GBFS approach on the binary dataset by considering multiple values of Num_trees = 100, 500, 700 and 1000 to identify the most common amongst all. From our observations, we conclude that mostly in each estimation the top 15 features remain the same.

We conclude, experimentally, that high accuracy values comply with a small learning rate, hence we decided to set the value of Num_trees = 1000 along with learning rate = 0.1. After computing 15 features of 15 sets of a binary dataset, we used the same features to compute the three-class, seven-class and multi-class dataset to detect the various attacks as all the four datasets of 15 sets have the same input measurement records - only the output label differs according to the category of the dataset. Table III represents the 15 features of 15 sets for all the four categories. Also, the primary goal of choosing a binary dataset to compute the promising features is to achieve faster execution speed and precise outcome in terms of detection rate as it only contains normal and attack vectors. Moreover, we have applied the same features to the rest of the three categories as basically all the categories are containing both malicious and normal events.

The datasets are well suited for ensemble classifiers since each set of the 15 datasets is produced at different attack locations by ORNL, and each consists of approximately 5500 records. The significance of the features depends on the location of the attacks on PMUs. Hence, the automatic stepwise feature selection is one of the crucial points for classification, which can be effectively handled by tree-based ensemble classifiers. Furthermore, for the proof of the concept, we have evaluated the accuracy of other machine learning techniques such as Naive Bayes, Support Vector Machine (SVM), Simple Logistic Regression (SLR), One Rule (OneR), Decision Table (DT) and Artificial Neural Network (ANN) for all the four categories as mentioned in Table III. We focus on tree-based ensemble classifiers since they give the best accuracy.

To evaluate the efficiency of the top 15 features in terms of detection rate and execution speed, we have applied various classifiers on all the 15 datasets of four categories. As the computed features are generated using the GBFS technique, we specifically target decision tree classifiers with a combination of boosting approaches such as GB, XGBoost, Random Forest(RF), AdaBoost Random Forest(AdaBoost-RF), ClassificationViaRegration- Random Forest(CVR-RF), Random Tree, AdaBoost Random Tree and J48.

Table IV
Comparative Analysis (accuracy) of various machine learning techniques

Classifiers	Naïve Bayes	SVM	SL	OneR	DT	ANN
Binary	54.17	70.2	78.04	81.87	89.86	88.34
Three-State	50.62	69.46	69.85	75.02	81.27	80.43
Seven-State	20.04	37.45	42.55	57.12	80.32	61.97
Multi-State	11.59	20.81	32.76	40.22	73.72	61.13

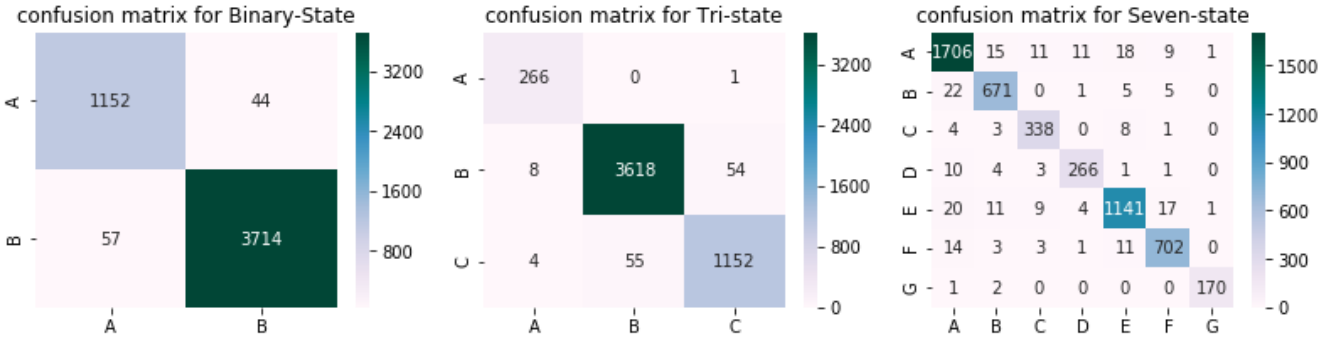
The proposed framework is programmed using Python on a Jupyter Notebook (Anaconda distribution) on Windows 10 with Intel Core i5-8300H 2.30GHz processor, 16 GB RAM and Nvidia Geforce GTX 1060 6go GPU. The results of classification of various classifiers are also validated using a WEKA platform [33]. The experiments are computed using random samples of 100,000 normal and attack observations for each of the four categories divided into 15 sets. The training and testing set of the model is obtained using 10-fold cross-validation methodology to measure the accuracy without biasing the normal or malicious output classes.

To assess the performance of each classifier, we have computed the following performance metrics: accuracy, detection rate, false-positive rate, F1 score and execution speed of 15 datasets of all the four categories. The results of performance metrics are derived from the confusion matrix during each classification. Figure 6 represents the example of one of the best confusion matrix of binary, three-class and seven-class classifier, respectively. Similarly, Figure 7 depicts the most promising confusion matrix of the multi-class classifier which can differentiate the total of 37 various attacks and normal events. By analyzing the confusion matrix, we can differentiate normal and attack vector in terms of True Positive, True Negative, False Positive and False Negative.

C. Result Discussion

The purpose of the proposed GBFS based feature selection framework is to generate a subset of the given attributes from entire dataset using a WFI metric to reduce the noise and improve the performance of the classifier. The derived subset of the top 15 features may or may not contribute same in the decision-tree classifiers. We have observed the results of total 8 decision tree-based machine learning techniques to validate our proposed methodology via multiple simulation trials. Overall 60 computations are performed to evaluate the performance of each classifier to include the results of fifteen datasets of all the four categories. Figure 7 represents the comparative analysis of the accuracy of eight decision tree-based classifiers of 15 datasets of each binary, three-class, seven-class and multiclass categories.

Amongst all the eight classifiers, it was observed that



(a) cf2 (b) cf3 (c) cf7

Figure 6: Confusion matrices, 2,3,7 output labels

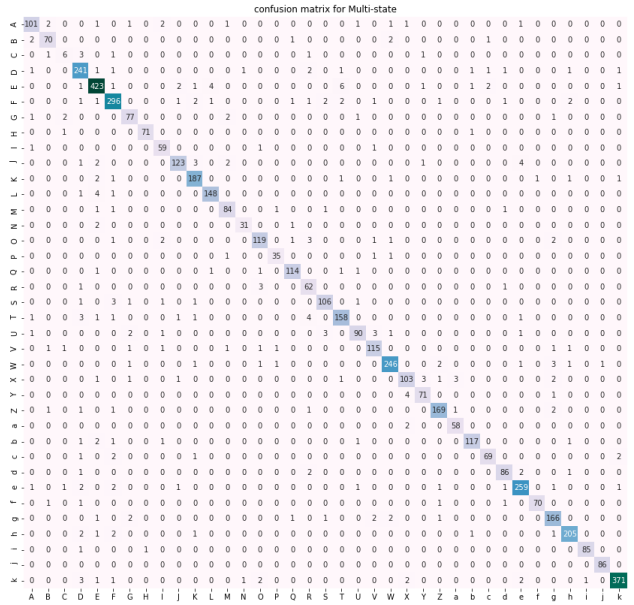


Figure 7: Confusion Matrix for the 37 output labels

XGBoost, random forest and its variance have proven to be most efficient. However, other tree-based classifiers also proved their efficiency ranging between 92 to 94 for Binary and three-state and 85 to 90 for seven class and multiclass. XGBoost comes up with accuracy equal to 97.26, 96.09, 92.97, 92.44 for binary, three-class, seven class and multiclass datasets, respectively. Similarly, all three variants of Random Forest also achieve very high accuracy such as 97.26, 97.24 and 97.17 for binary, 96.18, 96.38 and 96.50 for three-class, 94.43, 94.31 and 94.19 for seven class and 92.46, 92.92, 91.92 for multiclass, respectively. Since the GBFS-Random Forest and its variances are the most efficient classifiers to classify the normal and attack vectors with nearly same range of accuracy, we have compared the execution speed of all the three classifiers to identify the best among them. As depicted in Figure 9, GBFS-Random Forest classified the various attack and normal events for all the four categories in 1.5 seconds. GBFS-AdaBoost Random Forest took slightly more time than the GBFS-RF. GBFS-CVR-Random Forest took comparatively higher execution time as it uses the combined approach of boosting and ensemble of trees for the classification. However, by comparing the accuracy levels, we observe that the boosting does not much improve the result much, in such case GBFS-RF is proven to be best amongst all three with high accuracy and less execution time.

Table V

Performance evaluation metrics of Proposed GBFS Based Classifier

Measure	Binary	Three-class	Seven-Class	Multi-class
Accuracy	97.26%	96.50%	94.12%	92.46%
FPR	0.037	0.067	0.019	0.003
Precision	0.9705	0.9887	0.9504	0.9250
Recall	0.9740	0.9676	0.9355	0.9240
F-Measure	0.9723	0.9781	0.9427	0.9244

We demonstrated that the 15 stochastic features shown in Table III were the most promising features for all the decision tree-based classifiers by iteratively running all the eight classifiers, for 15 datasets of all the four categories. In each iteration, using 15 features, we retrained & re-tested all the eight decision-tree based models to compute the general average trend of malicious and normal events by observing DR, FPR and execution Time.

All the selected classifiers maintain very high DR and lower FPR rate in all the computations as shown in Table V. We have achieved 98.5% of detection rate which truly differentiates attack and normal vectors with only 3.7% and 6.7% of false positive rate for binary and three class classification. Moreover, seven-class and multi-class classifiers have also outperformed as they gave around 94.42% and 92.5% for the detection rate.

This validates the significance of our proposed methodology for feature selection. Real-time systems such as control and monitoring systems of industrial infrastructures/power grids need a methodology of feature extraction where processing time and storage space are always crucial.

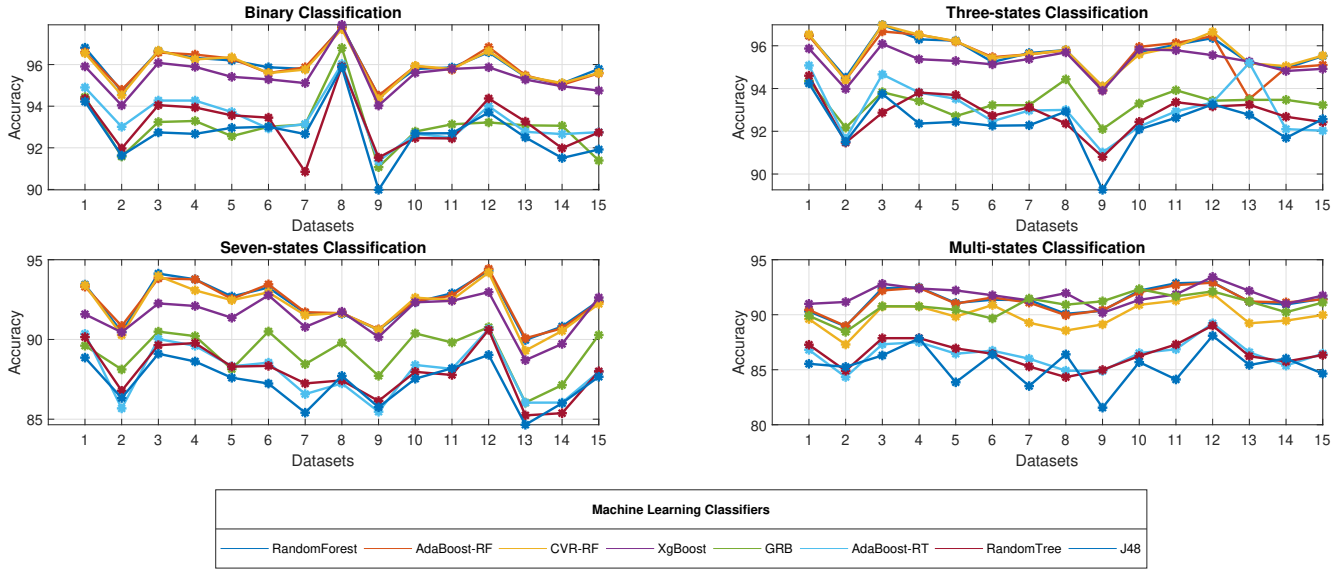


Figure 8: Comparative view of Different Machine Learning Classifiers for - four categories (binary, three-state, seven-state and multi-state) for each of 15 datasets

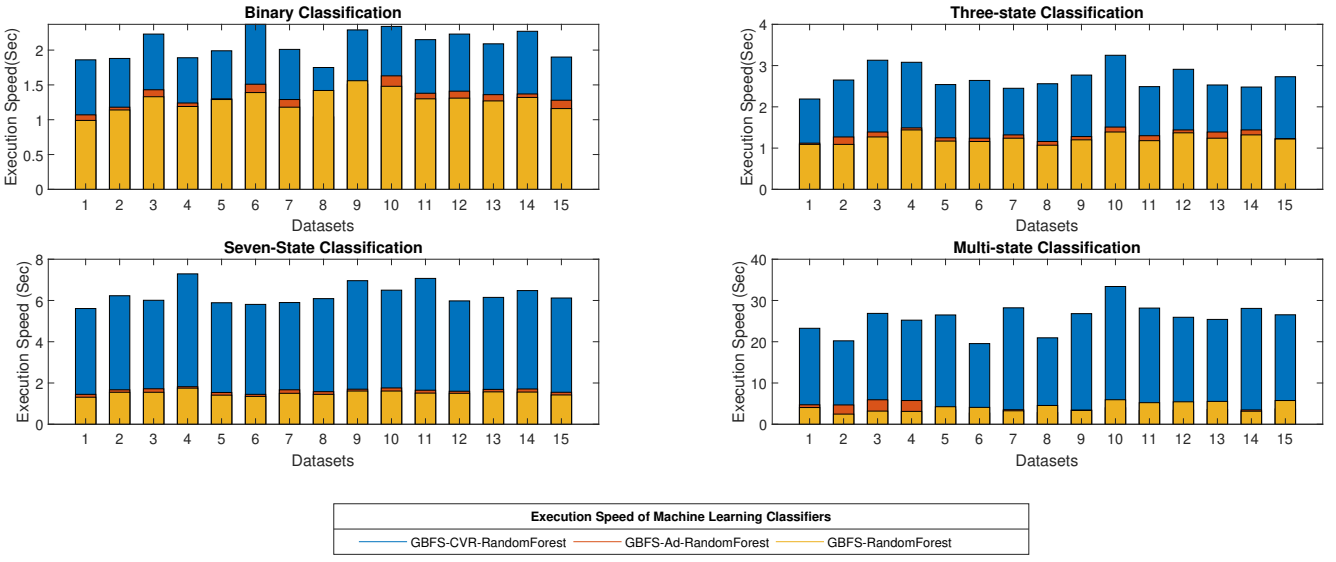


Figure 9: Comparative view of Execution speed of Three GBFS based Random Forest variances to classify normal and attack events for four categories (binary, three-state, seven-state and multi-state) for each of 15 datasets

To validate the efficiency of the proposed methodology, we have compared GBFS based decision tree algorithm with four published methods, namely AdaBoost-JRIP (AdaJRIP) [25], Common Path Mining [34], [28], Expectation Maximization Clustering Technique (EMCT) [32] and Gaussian Mixture – Kalam Filter Model (GMM-KF) using Pearson Correlation Coefficient (PCC) feature selection method [35], by considering various performance evaluation factors such as whether proper pre-processing is applied or not; to accelerate the process, whether feature selection approach is incorporated or not and if applied how many features are selected to evaluate the accuracy for various output classes.

It can be seen from Table VI that our proposed framework outperforms compared to those of the published techniques and accomplishes the highest accuracy with the 97.66%, 96.50% , 94.12% , 92.46% with only 12% of the features

for all the four categories of the power system datasets. Note that the results mentioned in the table refer to the highest accuracy achieved during the classification of the attacks and normal events by various tree based classifiers.

Table VI

Comparative analysis of overall performance of various techniques and Proposed GBFS Based Classifier

Classifier	Data Cleaning	Feature Selection	Features (%)	Classes	Accuracy
ADA-JRIP [25]	NA	NA	100%	2	94.55%
				3	94.61%
				37	85.85%
CPM [34], [28]	Applied	NA	100%	7	93.00%
				25	90.40%
EMCT [32]	Applied	PCC	25%	2	70.60%
			50%		76.3%
			75%		83.5%
			100%		90.2%
GMMKM [35]	Applied	PCC	25%	2	94.56%
			50%		95.83%
			75%		96.82%
			100%		97.27%
Tree Based	Applied	GBFS	12%	2	97.26%
				3	96.50%
				7	94.12%
				37	92.46%

Moreover, in order to show the efficiency, we have compared our proposed scheme with two well-known feature selection methods, namely, Chi-Square and Principal Component Analysis (PCA), in terms of the number of features, accuracy and execution time for a binary class using Random Forest (RF) classifier as shown in Table VII.

Table VII

Comparative analysis of various feature selection methods

Feature Selection Method	Classifier	Features	Class	Accuracy	Execution Time (sec)
Chi-Square	RF	106	2	96.69	4.6
PCA	RF	27	2	92.57	4.3
GBFS	RF	15	2	97.66	1.2

As mentioned earlier, data cleansing was performed to accelerate the process of classification using various machine learning algorithms. However, the technique in [25] has obtained comparatively low results with various well-known machine-learning algorithms such as OneR, SVM, Random Forest, Naive Bayes, JRIP and AdaBoost-JRIP owing to disregarding preprocessing before applying the classification approach on the power system dataset. As per our observations, the given dataset needs to be refined by removing infinite values before mapping and scaling the records. The features R1:PA:Z, R2:PA:Z, R3:PA:Z, R4:PA:Z, represent apparent impedance of the relay associated with IEDs of the given power system dataset comprising of infinite values and should be removed. However, in our proposed methodology, the top 15 features of any of the sets does not rely on impedance of relay attribute such as R1:PA:Z, R2:PA:Z, R3:PA:Z and R4:PA:Z. Hence we are essentially not deleting any row records of the given dataset.

Proper sanitization converts the type of the features from nominal to numeric which makes a huge impact in taking decision to classify the events of the given dataset by various classifiers. To demonstrate the impact of preprocessing and feature selection we have computed the results with and without preprocessing and with and without feature selection by applying all the eight decision-tree based classifiers on the power system dataset as mentioned in Table VIII.

The first two columns represent the accuracy and execution speed computed by eight decision-tree based classifiers without applying pre-processing on the dataset. In this case, all the classifiers have failed to achieve high accuracy and better execution speed because in order to predict the outcome, the classifier applies the modeling algorithm on both numerical and categorical inputs. At each iteration the decision-tree makes the decision by considering both the type of data in the dataset,that results in a long prediction time and low accuracy rate. Hence, proper sanitizing is the primary step for the classification.

Table VIII

Comparative analysis of various tree-based classifiers based on pre-processing and feature selection methodology

Algorithm	Cl	W/o Pre-Proc		W Pre-Proc		15 features	
		Acc (%)	Ex.sp Sec	Acc (%)	Ex.sp Sec	Acc (%)	Ex.sp Sec
XgBoost	2	71.14	8.14	96.23	3.46	97.21	1.98
GB	2	70.21	7.29	95.68	3.34	96.80	1.16
RF	3	72.66	8.53	95.01	5.35	96.18	1.47
ADA-RF	3	73.16	8.94	95.68	6.22	96.18	2.14
CVR-RF	7	59.22	9.98	93.20	8.36	94.42	5.14
J48	7	36.81	1.21	87.34	0.90	89.10	0.30
RT	37	27.73	0.35	88.25	0.11	90.01	0.04
ADA-RT	37	27.22	0.57	89.12	0.12	90.22	0.06

In contrast, the third and forth columns of the table represents the results computed by the eight classifiers by applying proper pre-processing on the entire dataset of 128 features. The pre-processing includes feature mapping, feature normalization and feature encoding techniques which improves the accuracy and execution speed.

Finally, we have combined pre-processing with feature selection to select the fifteen most promising features from the dataset before applying the classifier, which not only improves the accuracy but also improves the execution time. In a nutshell, our approach combines both pre-processing and feature selection, which has proven best amongst all the three approaches for all the decision-tree based classifiers.

VII. Conclusions and Future Work

This paper presented a GBFS based feature selection approach to identify the most promising features for anomaly detection in power grids. The overall framework consists of three key components. Initially, during data preprocessing,

the features are mapped and scaled to a specific range. To accelerate the execution speed and learning efficiency, a GBFS based feature selection approach is applied on filtered data to compute the most promising features from the entire dataset dynamically according to network/SCADA traffic. The dynamic approach of selecting the features from the entire dataset hides largely all the sensitive information of the power grid system. Finally, these reconstructed datasets are used by decision-tree based algorithms that classify the various attacks and normal events. The experimental results reveal the efficiency of the framework in terms of accuracy, detection rate, miss rate and execution speed compared to the original dataset. Moreover, the proposed GBFS based model outperforms some state-of-the-art techniques described in published works.

In the future, we plan to extend this work by combining the results of several classifiers to achieve an accurate outcome by applying majority vote ensemble method. This 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. We will target various 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. This approach will help to generate a better predicting model compared to a single model using "hard voting" based majority rule ensemble technique.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support in part by the Natural Sciences and Engineering Research Council (NSERC), Canada through a Collaborative Research Grant.

REFERENCES

[1] K. Poulsen, "Feature importance of feature selection," 2003, Accesed on 9/10/2019. [Online]. Available: https://www.securityfocus.com/news/6767

[2] 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

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

[4] "SANS and electricity information sharing and analysis center (e-isac). analysis of the cyber attack on the ukrainian power grid," Accesed 2019-09-28. [Online]. Available: http://www.nerc.com/pa/CI/ESISAC/Documents/E-ISAC_SANS_Ukraine_DUC_18Mar2016.pdf

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

[6] D. Upadhyay and S. Sampalli, "Scada (supervisory control and data acquisition) systems: Vulnerability assessment and security recommendations," *Computers & Security*, vol. 89, p. 101666, 2020.

[7] S. Nazir, S. Patel, and D. Patel, "Assessing and augmenting scada cyber security: A survey of techniques," *Computers & Security*, vol. 70, pp. 436 – 454, 2017.

[8] B. Barrett, "Security news this week: An unprecedented cyberattack hit us power utilities," Accesed 2019-11-14. [Online]. Available: www.wired.com/story/power-grid-cyberattack-facebook-phone-numbers-security-news/

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

[10] C.-C. Sun, A. Hahn, and C.-C. Liu, "Cyber security of a power grid: State-of-the-art," *International Journal of Electrical Power & Energy Systems*, vol. 99, pp. 45 – 56, 2018.

[11] C.-F. Tsai, Y.-F. Hsu, *et al.*, "Intrusion detection by machine learning: A review," *Expert Systems with Applications*, vol. 36, no. 10, pp. 11 994 – 12 000, 2009.

[12] Z. Xu, G. Huang, *et al.*, "Gradient boosted feature selection," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. New York, NY, USA: ACM, 2014, pp. 522–531. [Online]. Available: http://doi.acm.org/10.1145/2623330.2623635

[13] 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 Transactions on Computational Biology and Bioinformatics*, vol. PP, pp. 1–1, 07 2019.

[14] machinelearning, "Feature importance of feature selection," 2019, Accesed on 19.04.2019. [Online]. Available: https://machinelearningmastery.com/feature-importance-and-feature-selection-with-xgboost-in-python/

[15] F. Pedregosa, G. Varoquaux, *et al.*, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011. [Online]. Available: http://jmlr.org/papers/v12/pedregosa11a.html

[16] 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.

[17] R. Spenneberg, M. Brüggemann, and H. Schwartke, "Plc-blaster: A worm living solely in the plc," 2016. [Online]. Available: https://www.blackhat.com/docs/asia-16/materials/asia-16-Spenneberg-PLC-Blaster-A-Worm-Living-Solely-In-The-PLC-wp.pdf

[18] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 6th ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2013.

[19] S. J. Mousavirad, G. Schaefer, *et al.*, "High-dimensional multilevel maximum variance threshold selection for image segmentation: A benchmark of recent population-based metaheuristic algorithms," in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, ser. GECCO '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1608–1613. [Online]. Available: https://doi.org/10.1145/3377929.3398143

[20] Z. Zhu, Y.-S. Ong, and M. Dash, "Wrapper–filter feature selection algorithm using a memetic framework," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 1, pp. 70–76, 2007.

[21] P. M. Granitto, C. Furlanello, *et al.*, "Recursive feature elimination with random forest for ptr-ms analysis of agroindustrial products," *Chemometrics and Intelligent Laboratory Systems*, vol. 83, no. 2, pp. 83–90, 2006.

[22] F. Pan, T. Converse, *et al.*, "Feature selection for ranking using boosted trees," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ser. CIKM '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 2025–2028. [Online]. Available: https://doi.org/10.1145/1645953.1646292

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

[24] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 785–794.

[25] R. C. Borges Hink, J. M. Beaver, *et al.*, "Machine learning for power system disturbance and cyber-attack discrimination," in *2014 7th International Symposium on Resilient Control Systems (ISRCS)*, Aug 2014, pp. 1–8.

[26] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, pp. 1189–1232, 2000.

[27] T. H. Morris, Z. Thornton, and I. P. Turnipseed, "Industrial control system simulation and data logging for intrusion detection system research," in *7th Annual Southeastern Cyber Security Summit*, 2015, Huntsvile, AL. June 3 - 4, 2015.

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

[29] U. Adhikari, S. Pan, *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

[30] J. M. Beaver, R. C. Borges-Hink, and M. A. Buckner, "An evaluation of machine learning methods to detect malicious scada communications,"

in *2013 12th International Conference on Machine Learning and Applications*, vol. 2, Dec 2013, pp. 54–59.

[31] R. Punmiya and S. Choe, "Energy theft detection using gradient boosting theft detector with feature engineering-based preprocessing," *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 2326–2329, March 2019.

[32] M. Keshk, N. Moustafa, *et al.*, "Privacy preservation intrusion detection technique for scada systems," in *2017 Military Communications and Information Systems Conference (MilCIS)*, Nov 2017, pp. 1–6.

[33] I. H. Witten, E. Frank, *et al.*, *Data Mining, Fourth Edition: Practical Machine Learning Tools and Techniques*, 4th ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016.

[34] S. Pan, T. Morris, and U. Adhikari, "Classification of disturbances and cyber-attacks in power systems using heterogeneous time-synchronized data," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 650–662, June 2015.

[35] M. Keshk, E. Sitnikova, *et al.*, "An integrated framework for privacy-preserving based anomaly detection for cyber-physical systems," *IEEE Transactions on Sustainable Computing*, pp. 1–1, 2019.